

<u>조립 키트 매뉴얼</u>



KISCOM Co., Ltd.





목 차

1.	H/W	/ 소7	β4	ŀ
	1.1.	키트	. 구성물	ł
	1.2.	부품	·배치도	į
	1.3.	주요	.부품특징	5
	1.3.1	l.	콘덴서(Condenser)	5
	1.3.2	2.	저항기(Resistance)	5
	1.3.3	3.	코일(Inductor)	1
	1.3.4	1.	다이오드(Diode))
	1.3.5	5.	Diode의 활용10)
	1.3.6	5.	트랜지스터(Transistor)16	5
	1 4	1.1	지 하	2
	1.4.	기티	······································	,
2.	1.4. 아카	기티 데미	기트를 이용한 소프트웨어 제작25	;
2.	1.4. 아키 2.1.	기티 -데미 아키	지 않고	5
2.	1.4. 아카 2.1. 2.1.1	기티 - 테미 아키 l.	지 프로젝트 만들기	
2.	1.4. 아카 2.1. 2.1.1 2.1.2	기티 데미 아키 L. 2.	지 9	
2.	1.4. ●}₹} 2.1. 2.1.1 2.1.2 2.1.3	기티 아키 나. 2.	지 않 않 않 않 않 않 않 않 않 않 않 않 않 않 않 않 않 않 않	
2.	1.4. ○]-7] 2.1. 2.1.1 2.1.2 2.1.3 2.1.4	기티 -데미 아키 1. 2. 3.	지 않	
2.	1.4. •}₹ 2.1. 2.1.1 2.1.2 2.1.3 2.1.4 2.2.	기티 아키 나. 2. 3. 다일	지 3	
2.	1.4. •}₹] 2.1. 2.1.1 2.1.2 2.1.3 2.1.4 2.2. 2.2.1	기티 아키 아키 L. 3. 4. 파일	지 않 주 말 관리 프로그램 공통 함수	
2.	1.4. ○]-7] 2.1. 2.1.1 2.1.2 2.1.3 2.1.4 2.2. 2.2.1 2.2.1	기티 아키 아키 L. 3. 파일 L. 2.	지 않	





	2.3.1.	메뉴 작성 방법	32
2	.4. 데이	터 베이스 구성	34
	2.4.1.	도서 관리 프로그램	34
	2.4.2.	출석 관리 프로그램	35
2	.5. 도사	h 관리 프로그램 만들기	36
	2.5.1.	메인 폼 만들기	36
	2.5.2.	현재 장서 정보 프레임 만들기	37
	2.5.3.	책 정보 프레임 만들기	38
	2.5.4.	회원 정보 프레임 만들기	40
	2.5.5.	대여/반납 구분 프레임 만들기	42
	2.5.6.	파일 저장 프레임 및 대여 정보 리스트 만들기	43
	2.5.7.	도서 등록 폼 만들기	46
	2.5.8.	회원 등록 폼 만들기	52
2	.6. 출석	넉관리프로그램만들기	57
	2.6.1.	메인 폼 만들기	57
	2.6.2.	출석 현황 프레임 만들기	58
	2.6.3.	과목 선택 프레임 만들기	60
	2.6.4.	학생 정보 프레임 만들기	62
	2.6.5.	출석 현황 리스트 및 파일 저장 프레임 만들기	63
	2.6.6.	학생 등록 폼 만들기	67
	2.6.7.	과목 등록 폼 만들기	72





1. H/W 소개

1.1. 키트 구성물

내용물: 조립하기 전에 아래 테이블에 나열되어 있는 부품이 있는지 확인 한다.
 ※ 파란색으로 적혀 있는 부품은 보조 부품이므로 아래 테이블에 표시하지 않았음.

번호	Part	수량	참조	비고
1	0.1uF	11	C1,C2,C3,C4,C10,C13,C14,C15,C16,C17,C18	
2	15pF	3	C5,C6,C7	
3	47pF	2	C8,C9	
4	30pF	2	C11,C12	
5	10uF/25V	3	C19,C21,C22	
6	27pF	2	C20,C27	
7	130pF	2	C23,C25	
8	22pF	2	C24,C26	
9	LED	1	D1(LED Guide 포함)	Red
10	LED	1	D2(LED Guide 포함)	Green
11	Bridge Diode	1	D3	
12	DC Jack	1	J1	전원 포트
13	Buzzer	1	LS1	
14	2.2uH	2	L1,L2	
15	D-sub Right Angle Male	1	P1	통신 포트
16	330Ω	2	R1,R2	
17	10kΩ	6	R3,R4,R5,R6,R7,R8	
18	820Ω	1	R9	
19	2.7kΩ	1	R10	
20	4.7kΩ	1	R11	
21	Tack Switch	1	SW1	Reset Switch
22	KSD04H	1	S1	Baudrate 설정
23	74HC373	1	U1	
24	MFRC500	1	U2	RF Module
25	AT89C52(DIP)	1	U3(40Pin Dip Socket 포함)	Micom
26	HIN202CBN	1	U4	RS-232 Chip
27	7805	1	U5	Regulator
28	13.56MHz	1	Y1(Insulator 포함)	Crystal
29	22.1184MHz	1	Y2(Insulator 포함)	Crystal
30	Main P.C.B	1	P.C.B	92mm X 163mm
31	RS-232 Cable	1	Cross Cable(양쪽 Female)	통신 케이블
32	9V 300mA	1	Power Adaptor	전원





1.2. 부품 배치도







1.3. 주요 부품 특징

1.3.1. 콘덴서(Condenser)

1) 콘덴서의 개요

콘덴서란 전기를 축적하는 기능을 가지고 있다. 그러나, 일반적으로는 전기를 축적하는 기능 이외에 직류전류를 차단하고 교류전류를 통과시키려는 목적에도 사용된다. 회로도의 기호는 计 으로 표시한다.

콘텐서는 기본적으로는 2 장의 전극판을 대향시킨 구조로 되어 있다. 여기에 직류전압을 걸면, 각 전극에 전하(電荷)라고 하는 전기가 축적되며, 축적하고 있는 도중에는 전류가 흐른다. 축적된 상태에서는 전류는 흐르지 않게 된다. 10µF 정도의 전해콘텐서에 아날로그 미터식 테스터를 저항 측정 모드하고 접속하면 순간 전류가 흘러 테스터의 바늘이 움직이는 것을 알 수 있다. 그러나 바로 0 으로 되고 만다. 테스터의 접속 방법(콘텐서의 리드에 접속하는 테스터의 측정봉)을 반대로 하면 역시 순간 전류가 흐른다는 것을 알 수 있다. 그러므로, 직류전압이 콘텐서에 가해진 경우, 순간적으로 전류가 흐르지만 후에는 흐르지 않기 때문에 직류를 통과시키지 않으려는(직류 커트) 용도에도 사용된다. 그러나, 교류의 경우에는 앞서 언급한 테스터의 측정봉을 항상 교대로 바꾸어 접속하는 것과 같으므로 그 때마다 전류가 흐르게 되어, 교류전류는 흐르는 것이다.

두 극판의 전극간에 절연체(유전체라고 한다)를 넣어(절연체를 전극으로 삽입한다) 콘텐서를 만드는데, 이 재질에 따라 여러 종류의 콘텐서가 있다. 아무것도 삽입하지 않고 공기를 유전체로 하는 콘텐서도 있다. 콘텐서의 용량을 나타내는 단위는 패러드(farad: F)가 사용된다. 일반적으로 콘텐서에 축적되는 전하용량은 매우 작기 때문에, μF(마이크로 패러드: 10-6F)나 pF(피코 패러드: 10-12F)의 단위가 사용된다. 최근에는 슈퍼 커패시터라는 명칭으로 패러드 단위의 용량을 가진 콘텐서도 등장했다.

콘덴서의 용량 표시에 3 자리의 숫자가 사용되는 경우가 있다. 부품 메이커에 따라 용량을 3 자리의 숫자로 표시하든가, 그대로 표시하기도 한다. 3 자리 숫자로 나타내는 경우에는 앞의 2 자리 숫자가 용량의 제 1 숫자와 제 2 숫자이고, 3 자리째가 승수가 된다. 표시의 단위는 pF(피코 패러드)로 되어 있다.

예) 103 이면 10×10³ = 10,000pF = 0.01µF 224 이면 22×10⁴ = 220,000pF = 0.22µF 100pF 이하의 콘텐서는 용량을 그대로 표시하고 있다. 즉, 47 은 47pF 를 의미한다.

1.3.2. 저항기(Resistance)

1) 저항기의 개요

저항기란 전류의 흐름을 억제하는(흐름을 곤란하게 하는) 기능을 가지고 있다. 회로도의 기호로는 ------으로 표시한다. 저항값의 단위는 ohm(Q: 옴)이 사용된다. 그리고, 1000Q은 1kQ(킬로옴), 1000kQ은 1MQ(메가옴)이라 부른다. 저항기는 크게 고정저항기와 가변저항기로 나누어진다(이에 대해서는 나중에 설명한다). 그리고 사용하는 재료에 따라 탄소계와 금속계로 분류된다. 저항기를 사용하는 경우에 중요한 포인트는 저항값은 물론이거니와, 정격전력, 저항값 오차가 있다. 정격전력이란 저항기가 견딜 수 있는 소비전력(W: 와트)으로, 전력은 전류의 제곱(I2)×저항(R)으로 구할 수 있으며, 이 이하로 사용하지 않으면 저항기가 열을 발생하게 되고 결국 타버리는 경우도 흔히 있다. 전자회로에서 흔히 사용되는 것으로 1/8W, 1/4W, 1/2W 등이 있다. 전자회로의 신호회로(미약전류)에서는 너무 의식할 필요없이 1/8W 로 충분하지만, 전원회로, 발광 다이오드의 전류제어용과 같은 저항기에는 생각보다 큰 전류가 흐르기 때문에 정격전력을 염두에 둘 필요가 있다.

2) 정격전력

예를 들면 12V 의 전원전압을 사용하고, 5V 에서 동작하는 회로를 동작시키려는 경우를 생각해 보자. 이와 같은 경우, 보통은 3 단자 레귤레이터 등을 사용하지만, 간단히 저항기만으로 전압을 떨어뜨리려고 한 경우, 저항기의 저항값 이외에, 정격전력도 계산해 둘 필요가 있다. 이때, 5V 에서 동작하는 회로의 소비전류를 모르면 계산할 수 없다. 부품의 규격표에서 조사하거나, 시험 삼아 회로를 만들어 테스터로 측정해 보는 방법 등으로 구한다. 여기서는 그 소비전류가 100mA 였다고 하자. 저항값은 12V 에서 5V 로 낮추는 것이므로, 저항기에 7V 가 걸릴 필요가 있기 때문에 7V÷0.1A=70요가 된다. 이 저항기에서의 소비전력은 70요×0.1A×0.1A(또는 7V×0.1A)=0.7W 로 된다. 계산상으로 구해진 소비전력보다 여유가 있는 정격전력의 저항기를 선택한다. 이 경우 1W 가 적당할 것으로 생각한다. 기본적으로는 소비전력의 2배 정도에 해당하는 정격전력의 저항기를 사용하는 편이 무난하다.

> 한국폴리텍 IV 청주대학



1/8W 의 저항기로는 어느 정도의 전류를 취급할 수 있는지를 계산해 보면, 47kΩ 저항의 경우, √0.125W÷47KΩ=√2.66×10-6=1.63×10-3=1.63mA 로 된다. 전자회로의 신호회로에서 47kΩ에 이러한 전류가 흐르는 경우는 드물다. (흐를 때는 1.63mA×47kΩ으로 되어 76.6V가 걸리는 경우이다.)

3) 저항값

저항값의 표준에 대해서는 우리나라 KS 규격으로도 정해져 있지만, KS 규격은 그 대부분이 일본 JIS 규격을 모방하고 있으며, 실제로 업계에서는 JIS 규격을 훨씬 많이 이용하고 있다. 그래서 여기서도 JIS 를 기준으로 설명한다. 저항값의 표준은 JIS C5001 에서 E 표준 계열로 정해져 있다. 이것은 10 을 대수적으로 몇 등분하여 정해져 있다. 예를 들면 E3 의 경우, 10 을 대수적으로 거의 3 등분하여 [1], [2.2], [4.7], [10]으로 하고 있다. E6 의 경우는 [1], [1.5], [2.2], [3.3], [4.7], [6.8], [10]으로 된다. E12 는 [1], [1.2], [1.5], [1.8], [2.2], [2.7], [3.3], [3.9], [4.7], [5.6], [6.8], [8.2], [10]으로 된다. 저항값이 언뜻 보기에 제각기 무질서한 값으로 보이는 것은 이와 같은 이유 때문이다. E 계열은 3, 6, 12 이외에 24, 48, 96, 192 라는 계열이 있지만, 저항값의 표시는 숫자로 인쇄하기 위해서는 부품이 작기 때문에 컬러코드(color code)라고 하는 색깔로 표시하고 있는 경우가 많은데, 1/2W 이하의 저항기은 대부분 컬러코드로 표시하게 때문에 컬러코드를 읽는 법도 꼭 알아 둘 필요가 있다.



색깔	수치	비고
흑색	0	제 1, 2, (3)숫자, 승수
갈색	1	제 1, 2, (3)숫자, 승수, 오차
적색	2	제 1, 2, (3)숫자, 승수, 오차 30W
등색	3	제 1, 2, (3)숫자, 승수 60~300
황색	4	제 1, 2, (3)숫자, 승수 3
녹색	5	제 1, 2, (3)숫자, 승수
청색	6	제 1, 2, (3)숫자, 승수?
자색	7	제 1, 2, (3)숫자
회색	8	제 1, 2, (3)숫자
백색	9	제 1, 2, (3)숫자
금색	5%	오차
은색	10%	오차

1.3.3. 코일(Inductor)

1) 코일의 개요

코일이란 동선과 같은 선재(線材)를 나선 모양으로 감은 것이다. 회로기호는 **-*****으로 표시한다. 또, 코일의 성질 정도를 나타내는 단위로 헨리(Henry: H)가 사용된다. 선재를 감으면 감을수록 코일의 성질이 강해지며 헨리의 값도 커진다. 코일은 내부에 아무것도 넣지 않은 공심으로 하는 것보다 철심에 감거나 코어라 부르는 철분말을 응고시킨 것에 감는 편이 보다 큰 헨리값이 얻어진다. 통상 전기회로에서 사용하는 코일은 마이크로 헨리(µH)부터 헨리(H)까지 폭넓게 사용된다. 코일을 인턱터(Inductor) 또는 인덕턴스(Inductance)라고 하는 경우가 있다. (엄격히 말해서, 인덕턴스라고 하는 것은 코일 성분의 정도를 나타내는 것이며, 부품 그 자체를 나타내는 말이 아니다. 콘텐서의 경우는 커패시턴스(Capacitance), 저항의 경우는 레시스턴스(Resistance)라는 것이 각각 성질의 정도를 나타내는 것이다.)





코일에 교류전류가 흐른 경우, 코일에 발생하는 자속이 변화한다. 그 코일에 다른 코일을 가까이 했을 경우, 상호유도작용(Mutual Induction)에 의해, 접근시킨 코일에 교류전압이 발생한다. 이 상호유도작용의 정도를 상호 인덕턴스(단위는 헨리: H)로 표시한다. 코일이 하나만 있는 경우에도 자신이 발생하는 자속의 변화가 자신에게 영향을 준다. 이것을 자기유도작용이라고 하며, 그 정도를 자기 인덕턴스(Self Inductance)로 나타낸다. 헨리의 정의는 어떤 코일에 매초 1A 의 비율(1A/s)로 전류가 변화할 때, 다른 쪽의 코일에 1V 의 기전력을 유도하는 두 코일간의 상호 인덕턴스를 1 헨리(H)로 한다고 되어 있다. 자기 인덕턴스의 경우는 전류의 변화율이 1A/s 일 때 1V 의 기전력을 발생하는 경우의 자기 인덕턴스를 1H 로 한다고 되어 있다.

2) 코일의 성질

선재를 나선 모양으로 감으면 원래의 선재가 지닌 특성과는 전혀 다른 여러 가지 특성이 나온다. 여러 특성 가운데서 몇가지 주요 특성에 대해 그 개요를 쉽게 설명하기로 한다.



○ 전류의 변화를 안정시키려고 하는 성질이 있다.

전류가 흐를려고 하면 코일은 전류를 흘리지 않으려고 하며, 전류가 감소하면 계속 흘릴려고 하는 성질이다. 이것을 "렌츠의 법칙"이라 부르는데, 전자유도작용에 의해 회로에 발생하는 유도전류는 항상 유도작용을 일으키는 자속의 변화를 방해하는 방향으로 흐른다는 것이다.

이 성질을 이용하여 교류로부터 직류로 변환하는 전원의 평활회로에 사용된다. 교류를 정류기에 의해 직류로 변환한 경우, 그대로는 맥류(리플: Ripple)라고 하여 교류성분이 많은 직류이며 완전한 직류가 아니다. 플러스의 직류로 정류한 경우, 마이너스 전압성분은 없어지지만, OV 와 플러스 전압을 왕래하고 있다. 평활회로는 콘텐서와 코일을 조합한 회로를 사용하면 코일이 전류의 변화를 저지하려는 작용을 하고, 콘텐서가 입력전압이 OV 로 되어도 축적한 전기를 그때 토출하기 때문에 안정한 직류를 얻을 수 있다. 간단한 평활회로에서는 코일 대신에 저항기를 사용하여, 콘텐서의 평활 기능만 이용하는 경우도 있다.

○ 상호유도작용이 있다.



이것은 앞에서도 언급했지만, 두 코일을 가까이 하면 한쪽 코일의 전력을 다른 쪽 코일에 전달할 수 있다는 것이다.이 성질을 이용한 것이 트랜스이다. 전력을 공급하는 쪽의 코일(입력)을 1 차측, 전력을 꺼내는 쪽(출력)을 2 차측이라고 한다. 1 차측 권수와 2 차측 권수의 비율에 따라 2 차측의 전압이 변화한다. 전원트랜스 등은 2 차측에서 권선의 도중에 선을 내어(tap 이라고 한다) 복수의 전압을 얻을 수 있도록 한 것이 많다.

○ 전자석의 성질이 있다.



여러분이 잘 알고 있듯이, 전류가 흐르면 철이나 니켈을 흡착하는 성질이다. 이 성질을 이용한 것으로 계전기(릴레이)가 있다. 전류가 흐를 때에 철판을 끌어당겨 철판에 부착된 스위치를 닫도록 하는 것이다. 그리고 차임벨도 전자석의 성질을 이용한 것이다.

○ 공진하는 성질이 있다.





코일과 콘덴서를 조합하면 어떤 주파수의 교류전류가 흐르지 않거나, 쉽게 흐르기도 한다. 라디오의 방송국을 선택하는 튜너는 이 성질을 이용하여 특정한 주파수만을 선택하고 있는 것이다. 공진(共振)에 대하여 설명하기 시작하면 점점 깊은 부분까지 빠져들기 때문에 더 자세하게 알고 싶은 분은 전문서를 보기 바란다.



1.3.4. 다이오드(Diode)

1) 다이오드의 개요

다이오드란 전류를 한쪽 방향으로만 흘리는 반도체 부품이다. 반도체란 원래 이러한 성질을 가지고 있기 때문에 반도체라 부르는 것이다. 트랜지스터도 반도체이지만, 다이오드는 특히 이와 같은 한쪽 방향으로만 전류가 흐르도록 하는 것을 목적으로 하고 있다. 반도체의 재료는 실리콘(규소)이 많지만, 그 외에 게르마늄, 셀렌 등이 있다. 다이오드의 용도는 전원장치에서 교류전류를 직류전류로 바꾸는 정류기로서의 용도, 라디오의 고주파에서 신호를 꺼내는 검파용, 전류의 ON/OFF 를 제어하는 스위칭 용도 등, 매우 광범위하게 사용되고 있다. 회로기호는



기호의 의미는 (애노드) —▶ (캐소드)로 애노드측에서 캐소드측으로는 전류가 흐른다는 것을 나타내고 있다. 다이오드 중에는 단지 순방향으로 전류가 흐르는 성질을 이용하는 것 이외에, 다음과 같은 용도의 것이 흔히 사용된다.

○ 정전압 다이오드(제너 다이오드)

회로 기호는 — 이다. 역방향으로 전압을 가했을 경우에 어떤 전압에서 안정하는 성질을 이용하여, 일정한 전압을 얻기 위해 사용한다.

○ 발광 다이오드(LED)

회로 기호는 ━━━━━ 이다. 전류를 순방향으로 흘렸을 때에 발광하는 다이오드이다.

○ 가변용량 다이오드(배리캡 또는 버랙터: varactor)

회로 기호는 📑 이다. 전압을 역방향으로 가했을 경우에 다이오드가 가지고 있는 콘덴서 용량(접합용량)이 변화하는 것을 이용하여, 전압의 변화에 따라 발진주파수를 변화시키는 등의 용도에 사용한다. 역방향의 전압을 높이면 접합용량은 작아진다.







위의 그래프는 다이오드의 특성을 나타낸 것이다. 순방향으로 전압을 가했을 경우, 약간의 전압에서도 순방향의 전류는 쉽게 흐른다는 것을 나타내고 있다. 순방향으로 흘릴 수 있는 전류는 다이오드에 따라 규정되어 있다. 그리고 통상적으로 사용하는 경우 다이오드 자체의 저항성분에 의해 강하하는 전압은 0.6~1V(VF) 정도이다(실리콘 다이오드의 경우, 대략 0.6V). 여러 개의 다이오드를 직렬로 접속하여 사용하는 회로에서는 이 전압강하도 고려할 필요가 있다. 정류용으로 사용하는 경우, 순방향의 전류 허용값은 중요한 체크 포인트이다. 역방향으로 전압을 가했을 경우, 역방향 전류는 흐르기 어렵다는 것을 나타내고 있다. 역방향으로 가할 수 있는 전압은 다이오드의 종류에 따라 여러 가지가 있으므로 용도에 따라 선택한다. 그리고 역방향 전류는 매우 작어 수 μA 에서 수 mA 이며, 다이오드의 종류에 따라 다르다. 정류용으로 사용하는 경우, 역방향의 전압 허용값은 중요한 체크 포인트이다.

조립 키트 매뉴얼

1.3.5. Diode 의 활용

1) 정류

다음은 다이오드를 이용하여 교류(AC) 전압을 직류(DC) 전압으로 변환시키는 과정을 살펴보자.

(1) 교류와 직류

이에 앞서 교류와 직류의 차이에 대해 살펴본다.

발전소에서 만들어지는 전기는 매우 높은 전압의 교류인데 이것이 몇 단계의 변압과정을 거쳐 우리에게 최종 220V 의 교류로 공급된다. 교류는 시간의 흐름에 따라 반복적으로 전압이 + (plus)와 - (minus) 로 변한다. 이와 같은 교류의 흐름을 시간 개념과 함께 그림으로 표현하면 다음과 같다.



<그림 1-1 교류의 모양>

이에 비해 직류(Dicrect Current, DC)는 항상 일정한 수준의 전압을 유지한다.



10

Rev.1.0



조립 키트 매뉴얼



<그림 1-2 직류의 모양>

<그림 1-2>는 + (plus)인 전압을 갖는 직류를 나타냈지만 일정하게 - (minus)인 경우도 있다.

전기를 동력으로 하는 기기들은 모타나 전등과 같이 교류로 잘 동작이 되는 것도 있지만 오디오앰프는 구성 소자들의 작동을 위해 직류를 사용한다. 직류는 미리 직류를 저장한 배터리로도 공급받을 수 있다. 다만, 오디오 앰프는 특히 출력이 높을수록 많은 전기를 소모함으로, 일반적으로는 헤드폰 앰프와 같이 소규모 앰프 정도만이 배터리를 이용한다. 배터리는 이론적으로는 보다 깨끗하고 직선적인 직류를 공급하기 때문이다.

이에 비해 전원장치와 같이 교류를 직류로 바꾸는 과정에서는 충분히 직류로의 변환이 되질 않아 약간의 맥류(ripple)가 남거나, 직류의 흐름에 교류성분이 유도되어 들어오는 경우가 있기 때문에 배터리와 같이 깨끗한 직류를 만들어내기 위해서는 상당한 노력이 필요하다.

(2) 정류의 원리

교류를 직류로 변환하는 것을 정류(rectify)라고 한다. 교류를 직류로 바꾸기 위해서는 (+)와(-)의 전압이 교대로 나타나는 교류의 흐름 중에서 (+) 의 전압만 걸러내면(+)의 전압을 갖는 직류를 얻을 수 있다. 만약(-)의 직류 전압이 필요 하다면 교류(-)의 전압만을 걸러 내면된다.

(+) 혹은(-)의 전압만 걸러내는 역할은 다이오드(diode)가 수행한다. 다이오드는 반도체 성질을 이용하여 만든 소자(device)로, 한 방향으로만 전류가 흐르도록 하는 특성을 지니고 있다. 마치 좁은 골목길을 일방통행으로 정하듯이, 순방향은 통과, 역방향은 금지의 교통정리를 해준다.

(3) 다이오드의 외관

다음 <그림 1-3>은 전원부에 들어가는 단일 다이오드와 단일다이오드 네개를 하나로 몰딩한 브릿지 다이오드들의 모습이다. 브릿지 다이오드도 그 크기에 따라 사용전압, 전류를 달리한다.

그림의 가장 작게 보이는 다이오드는 전원부가 아닌 신호부분에서 사용되는 것이다.



<그림 1-3 여러 가지 다이오드>





(4) 회로도 기호

회로도에서 다이오드는 아래<그림 1-4>와 같은 기호로 표시한다.



━━━ 순방향

<그림 1-4 회로도 상의 다이오드 기호>

다이오드는 + 극인 anode(약자:a)와 -극인 cathode(약자:k)의 두개의 극을 갖고 있다. <그림 1-4>에 표시한 방향성은 anode 가 cathode 보다 높은 전압일 때 순방향임을 의미하며, 다이오드는 순방향으로는 전류를 통과 시키지만 역방향이면 전류를 흐르지 못하게 차단하는 기능을 갖고 있다.

다이오드를 보면 <그림 1-5>와 같이 띠가 표시되어 있는데, 이것은 (-)극인 cathode 극을 의미한다.



<그림 1-5 다이오드의 극성 표시>

다이오드는 극성을 틀리게 연결하면 엉뚱한 결과가 나오던지, 다이오드가 파손되므로 조심하여야 한다.

(5) 정류 방법

가. 반파(Half Wave) 정류

트랜스로 변압된 교류를 다이오드 1 개로 정류하는 방법이다.



<그림 1-6 반파 정류회로>

위의 회로는 다이오드의 특성을 이용하여 (+)의 교류 전압만 걸러내는 회로이다. 다이오드를 통과한 직후의 직류의 모습은 아래 <그림 1-7>와 같다. 아직은 직류라고는 하지만 울퉁불퉁한 직류형태이다.



<그림 1-7 반파 정류한 결과>

반파정류는 정류결과가 거칠고 효율성도 떨어지므로 아주 단순한 회로 이외에는 잘 사용되질 않는다.

나. 전파(Full Wave) 정류

다이오드 2개를 이용, 교류가 +/- 어느 상태든 정류를 하는 방법으로 회로구성은 <그림 8>과 같다.









<그림 1-8 전파정류>



전과정류는 반과정류에 비해 효율이 높은 대신 트랜스의 2 차 전압을 반으로 나누어 쓰기 때문에 반파정류와 같은 전압을 만들려면 반파정류에 비해 전압이 두 배 높은 트랜스를 사용하여야 한다.

다. 브릿지(Bridge)방식의 정류

다이오드 4 개를 이용, 교류의 +/- 전압 어느 상태든 모두 걸러내는 정류 방법으로 <그림 10>과 같은 회로로 구성된다.



<그림 1-10 브릿지 방식의 정류>

브릿지 방식은 교류의 +/- 양 전압을 모두 걸러 사용하는 면에서는 전파(Full Wave) 정류와 같지만, 전파 정류는 트랜스의 2 차측을 반으로 나누어 직류를 만드는데 비해 브릿지 방식은 2 차측 전체전압을 직류로 변환하기 때문에 전파 정류방식에 비해 2 배 가까운 전압을 얻을 수 있는 점이 다르다.



브릿지 방식은 투입비용에 비해 효율이 높고 안정된 정류결과를 얻을 수 있기 때문에 오디오 앰프의 전원은 거의 이 브릿지 방식을 취하고 있다.

(6) 다이오드의 규격





다이오드의 규격에는 여러 특성이 표기 되는데, 그중 중요한 것은 그 다이오드가 견딜 수 있는 최대 전압치와 최대 전류치이다. 전원에 사용하는 다이오드는 안전과 발열을 고려, 실제 사용 전압/전류의 2~3 배 정도 여유를 둔다.

2) 평활(Smoothing)

(1) 평활의 의미와 원리

다이오드에 의해 교류에서 변환된 직류는 아직 깨끗하지 못한 형태이다. 배터리와 같이 보다 이상적인 직류와 비교하면 다음과 같다.



<그림 2-1 다이오드로 정류한 직후의 직류>

따라서 보다 직선적인 직류 형태로 만들기 위해 평활콘텐서를 통과시키는 과정을 갖게 된다.





평활 콘텐서는 아파트 옥상에 있는 물탱크처럼 전기를 비축해두는 탱크라는 이미지를 연상하면 된다. 즉, 탱크에 유입되어 들어오는 들쑥날쑥한 직류의 전압높이를 이 탱크에 잠재워 평탄한 직류로 만드는 것이다. 콘텐서에 비축되는 직류는 다음과 같은 형태이다.



<그림 2-3 평활 콘텐서에 의한 평활후의 직류모습>





위의 그림은 맥박처럼 뛴다는 의미의 맥류(ripple)가 과장된 형태로 표현되어 있는데, 일정수준까지는 콘덴서의 용량이 클수록 더욱 평탄한 직류를 얻을 수 있다.

(2) 평활용 콘덴서의 외관

콘덴서는 영어권에서는 주로 캐퍼시터(Capacitor)라고 표현하기도 하는데, 전기를 비축할 수 있는 용량이 작은 것부터 큰 것까지, 또한 사용하는 용도에 따라 다양한 종류의 콘덴서가 있다. 그 중에서 전원장치의 평활부분에 사용하는 콘덴서는 용량이 큰 것이 필요하므로 큰 용량을 손쉽게 얻어낼 수 있는 전해콘덴서(electrolytic condenser)를 사용한다.

아래<그림 2-4>는 각종 전해 콘텐서를 비교해서 찍은 것이다.



<그림 4 의 4 각종전해콘덴서>

가장 오른쪽의 것이 100W 정도의 파워앰프에 장착하는 전해 콘덴서이고, 그 다음이 프리앰프 전원부용 콘덴서이다. 나머지는앰프부분의회로나기타부분에사용되는소형들이다. 전해콘덴서는 사용할 수 있는 전압이 높을수록, 용량이 커질수록 외형이 급격하게 커져서, 외형이 큰 것은 주먹만한 것도 있다.

(3) 회로도 기호

회로도에서 전해콘덴서 기호는 여러 나라의 관습상 약간씩 달리 그려왔는데 기호모양이 좀 다르더라도 전해콘덴서 기능은 다 같은 것이다.



<그림 4 의 5 전해콘덴서의 기호>

전해콘텐서는 +/- 의 전압극성이 있어 제품의 두 가닥 리드선의 극성이 표시되어 있다. 회로에서 콘텐서를 배치할 때에는 반드시 이 극성대로 연결 하여야 한다. 예를 들어 콘텐서를 연결할 두 점 중, 한 점이 +5V 이고 다른 점이 +2V 라면 콘텐서의 (+)극을 +5V 점에 연결하여야 한다. 만약 거꾸로 연결하면 열이 나면서 안에 들어있는 용액이 흘러나오거나, 심하면 겉의 알루미늄 표피가 터지면서 용액이 쏟아져 나오기 때문에 회로를 구성할 때, 제대로 연결했는지 반복하여 확인해야 한다.

(4) 전해콘덴서의 규격

전해콘덴서의 규격 중 중요한 것은 그 전해콘덴서가 견딜 수 있는 전압(내압)과 용량이다. 모든 전해콘덴서는 항상 실제 배치된 곳의 전압보다 높은 내압의 것을 사용한다.

한국폴리텍 IV 청주대학

15

Rev.1.0



전해콘덴서의 내압은 12V, 15V, 25V, 35V … 등과 같이 업계 표준내압만 있으므로 예를 들어 콘덴서의 양극에 실제 22V 의 전압이 인가된다면 내압이 25V 나 35V 의 제품을 택한다.

한편, 전원장치의 평활용 전해콘덴서는 용량이 클수록 평활의 효과가 크지만, 무한정 큰 것을 채택할 수도 없고, 일정용량이 상이 되면 그 효과도 정비례하지 못하고 점증하는 경향이 있으므로 적당한 선에서 결정해야 한다.

3) Bridge Diode 의 활용

- 본 Academy Kit 에서는 Bridge Diode 의 특성을 이용하여 DC 전압이 거꾸로 들어와도 항상 단말기 단의 전원은 (+)극을 유지할 수 있어서 단말기에 올바른 전원을 공급할 수 있다.

- 참고로 본 Academy Kit 에 사용된 Bridge Diode 는 1A 의 전류까지 견딜 수 있다.

1.3.6. 트랜지스터(Transistor)

1) 트랜지스터의 개요

트랜지스터는 기본적으로는 전류를 증폭할 수 있는 부품이다. 아날로그 회로에서는 매우 많은 종류의 트랜지스터가 사용되지만, 디지털 회로에서는 그다지 많은 종류는 사용하지 않는다. 디지털 회로에서는 ON 아니면 OFF 의 2 치신호를 취급하기 때문에 트랜지스터의 증폭 특성에 대한 차이는 별로 문제가 되지 않는다. 회로 기능은 대부분이 IC 로 처리하는 경우가 많다. 디지털 회로에서 트랜지스터를 사용하는 경우는 릴레이라고 하는 전자석 스위치를 동작시킬 때나 (릴레이는 구동전류를 많이 필요로 하기 때문에 IC 만으로는 감당하기 어려운 경우가 있다.) 발광 다이오드를 제어하는 경우 등 이다.

회로 기호는 PNP 타입은 _____, NPN 타입은 _____ 으로 표시한다. 트랜지스터는 반도체의 조합에 따라 크게 PNP 타입과 NPN 타입이 있다. 그리고, 트랜지스터는 용도와 상기의 타입에 따라 다음과 같은 명칭이 붙여진다.

- ▶ 2SA×××□□□□□□□□□PNP 타입의 고주파용
- ▶ 2SB×××□□□□□□□□□□PNP 타입의 저주파용
- ▶ 2SC×××□□□□□□□NPN 타입의 고주파용
- ▶ 2SD××× □□□□□□□NPN 타입의 저주파용

PNP 타입과 NPN 타입에서는 전류의 방향이 다르다. 마이너스 전압측을 접지로, 플러스 전압측을 전원으로 하는 회로의 경우, NPN 타입 쪽이 사용하기 쉽다.



○ 트랜지스터의 외관



트랜지스터의 외관은 여러 가지가 있지만, 여기서는 가장 일반적인 두 종류만 실었다.사진에서 좌측의 트랜지스터는 2SC1815 이라는 것으로, 디지털 회로에서 흔히 사용한다. 여러 용도에 사용할 수 있으므로 몇 개씩 낱개로 구입하기보다는 200 개가 들어 있는 팩으로 구입하면 경제적이다. 사진에서 우측의 트랜지스터는 2SD880 으로, 큰 전류를 취급하려는 경우에 사용한다.

전기적 특성은 각각 아래와 같다.





조립 키트 매뉴얼

항 목	2SC1815	2SD880	
V _{CEO} (V)	50	60	
l _c (mA)	150	3A	
P _C (mW)	400	30W	
h _{FE}	70~700	60~300	
f⊤(MHz)	80	3	

○ VCEO: 베이스(B)를 오픈했을 때에 컬렉터(C)와 이미터(E)에 걸리는 최대전압.

(단순히 VCE 로 표시하는 경우도 있다)

○ IC: 최대 컬렉터(C) 전류.

○ PC: 주위온도(Ta)=25℃에서 연속해서 소비시킬 수 있는 최대 컬렉터(C) 손실(방열기 없음)

- hFE: 이미터(E) 접지에서의 직류에 대한 전류증폭률(IC÷IB).
- fT: 주파수를 높여가면 증폭 능력이 저하하는데, 베이스 전류(IB)와 컬렉터 전류(IC)가 같아지는 주파수. [직류증폭을 할 수 없게 되는 주파수(트랜지션 주파수)]

2) 트랜지스터의 리드



트랜지스터의 종류에 따라 리드의 내용이 다르기 때문에 매뉴얼 등을 참조하여 확인할 필요가 있다. 2SC1815 의 경우 품명이 인쇄되어 있는 평평한 면을 바라 보았을 때, 오른쪽 리드가 베이스중앙의 리드가 컬렉터왼쪽의 리드가 이미터이다. 2SD880 의 경우 품명이 인쇄되어 있는 면을 바라 보았을 때,오른쪽 리드가 이미터중앙의 리드가 컬렉터왼쪽의 리드가 베이스이며, 2SC1815 와는 반대이다.





1.4. 기타사항

◎ 납땜시 준비물

인두, 인두받침, 젖은 스티로폼(젖은 걸레나 휴지도 무방), 땜납, 니퍼, PCB

◎ 납땜 인두의 사용 방법

전자 키트 조립에 사용하는 납땜 인두는 정격 소비 전력이 15~30W 정도의 것이 가장 알맞다. 권총형 납땜 인두를 사용하면 반도체로 구성된 전자 부품이 과도한 유도 전류에 의하여 손상되는 경우가 있으므로 사용하지 않는 것이 좋다. 인두 끝은 항상 깨끗헤게 유지해야 하며, 인두 끝의 온도가 너무 높으면 전자 부품이 손상되거나 인쇄회로의 구리박이 들뜨기 쉽다. 인두 끝의 길이를 길게 조절하면 인두끝의 온도를 약간 낮출 수 있다.

◎ 인두는 적당한가?

납땜을 할 때는 전기인두를 사용합니다. 적정온도는 400-450 도이고, 인두 끝은 까맣게 산화되거나 납이 잘 녹지 않거나 녹았을 때 윤기가 없다면 온도가 너무 높거나 낮은 것입니다. 전기인두의 끝은 항상 청결하게 한 후에 사용하시는 것이 좋습니다. 가열하기 전에 깨끗이 닦아놓는 것이 좋다. 어떤 인두의 팁은 산화방지를 위해 코팅을 해놓은 것이 있는데 이것은 그냥 휴지에 물을 묻혀서 닦은 다음 사용하면 된다.

◎ 땜납은 좋은가?

일반적으로 땜납은 주석과 납이 일정비율로 섞여 있다. 그 비율에 따라 납이 잘 녹거나 잘 녹지 않기도 한다. 주석이 많으면 납은 잘 녹지는 않지만 잘 붙는다. 납이 많으면 그 반대의 성질을 보이는데 많이 사용되는 땜납은 주석과 납의 비율이 40% : 60% 이다. 그밖에 은이 소량 첨가된 은납이라는 것을 사용하기도 하는데 은이 첨가되어 전도성이 우수하다고 하지만 은의 함유량은 상당히(수입 은납의 경우 2% 정도) 적고 값은 비싸므로 굳이 은납을 사용할 필요는 없다.(가격이 일반납의 5 배정도) 그보다도 더 중요한 것은 땜납의 굵기이다. 너무 굵은 땜납은 잘 녹지도 않고 납땜의 크기도 상당히 커진다. 눈으로 보아서 가늘다고 생각되는 것이 납도 잘 녹고 납땜의 양을 조절하는데도 편리하다.

◎ 가열시간은 적당히

납땜을 하고자 하는 모재(기판이나 부품)의 재질에 따라서 가열시간은 차이가 있지만 대체로 모재가 열을 잘 흡수하면 충분히 가열하고, 모재가 작거나 열을 잘 흡수하지 않는 재질이면 대략 2 초정도가 적당하다.

◎ 납땜의 경사각과 납량

납땜을 할 때 초보자가 가장 많이 범하는 실수는 납땜의 양이 많아진다는 것이다. 납량이 많아지면 납땜의 경사각이 커지게 된다. 납이 기판의 표면이 잘 밀착이 되도록 하는 것이 중요하며 납을 많이 사용하여 납이 동그랗게 뭉치지 않도록 해야 한다.

◎ 납땜하는 순서

- ① 우선 인두팁과 납을 납땜할 곳에 가까이 한 후, 납땜할 곳을 확인한다.
- ② 모재에 인두팁을 대서 모재를 가열합니다. 가열시간은 위에서 언급한 것과 같이 한다.
- ③ 모재에 납을 댑니다. 가열된 모재에 납이 녹는다.
- ④ 납이 녹는 정도를 보아가며 납을 댄다.
- ⑤ 인두를 땝니다. 인두를 땔 때는 방향과 속도에 주의한다. 너무 빨리 때면 납땜이 동그랗게 되지 않는다.

◎ 후처리

납땜이 끝나면 납땜이 된 곳 부근을 깨끗이 한다. 납땜을 용이하게 하기 위해서 플럭스(왁스처럼 생긴 유기용제)를 쓰기도 하는데 이것을 사용한 경우에는 납땜의 주변에 끈적끈적한 찌꺼기가 남게 된다. 이는 TC 용액으로 닦아내면 깨끗해진다. 하지만 이를 닦아내지 않을 경우 간혹 납 찌꺼기나 전선 부스러기 등이 붙어 쇼트의 원인이 되기도 한다.

◎ 실패한 납땜을 고치는 방법

- ① 납이 부족한 경우는 인두로 열을 가하면서 한 번 더 납땜을 한다.
- ② 실패한 납땜을 뽑아내고 한 번 더 납땜을 한다. 납땜을 뽑아내는 도구는 납 흡입선 및 납 흡입기 등을 사용한다.
- (납 흡입기를 이용하여 납땜을 뽑아낼 경우 PCB 에 손상을 줄 수도 있음)

◎ 납땜 시 주의사항

- ① 납땜인두를 기판 한곳에 너무 오래 대고 있으면 기판이 들뜨거나 반점이 생기기도 한다.
- ② 트랜지스터나 IC 등 열에 약한 부품을 납땜할 때에는 롱노우즈 플라이어나 핀셋으로 다리를 잡아주어서 열이 부품내부로 집중되는 것을 막아주어야 한다. 또한 오래 가열하지 말고 신속하게 납땜을 해야 한다.
- ③ 마지막으로 납땜인두는 매우 뜨거우므로 납땜이 끝나면 반드시 전원플러그를 빼두고, 인두는 전용스탠드나 사기 등으로 된 받침대 위에 확실히 올려놓아야 한다. 인두가 방바닥 위에 떨어진 상태로 방치를 하게 되면 화재가 발생할

한국폴리텍 IV 청주대학



수도 있다.

◎ part number S1 의 Dip type switch 의 용도 - baud rate 조정

Academy Kit 에 있는 4Pin Dip Switch 는 Baudrate 를 테스트할 수 있는 부품으로서 아무것도 설정되지 않은 상태는 표준 상태로서 38,400 BPS 통신을 한다. 아래의 그림은 각각의 통신 속도에 맞는 Dip Switch 를 표시한 그림이다.



** baud rate - 데이타 전송 속도의 측정치. 보통 초당 비트(bits)수로 나타낸다. 컴퓨터 또는 통신장비 간에 데이터 흐름의 양을 측정하기 위해서 사용된다.

◎ 안테나 Reading Range 조절

Capacitor C20, C23, C27, C33 의 값을 조정하여 안테나의 Reading Range 를 조절 할 수 있다. 현재 공급된 Reader 의 Range 는 약 5 Cm 이내이다. (최대 10 Cm 까지 가능하다.)

◎ 아래 주의사항 참조)

◎ 주의 사항

① part number LS1, D1, D2, D3, C21, C22 의 경우 극성에 주의하며 납땜한다.

- ② part number U1, U2, U3, U4 의 경우 PCB 의 오목한 부분과 칩의 오목한 부분을 맞춰 납땜한다. : 디바이스의 방향이 반대로 되면, 전원이 쇼트상태가 되기 때문에 디바이스가 파손될 우려가 있다.
- ③ part number C20, C23, C25, C27 의 경우 SMD 타입이므로 납땜 시 주의한다.
- ④ P.C.B 에 나타나 있는 각각의 Component 들의 패턴은 인두기의 열에 의해 떨어질 수 있으므로 가급적 납땜 시 장시간 열을 가하지 않는다.
- ⑤ 납땜 시 SMD Type 의 부품부터 한다.
- : 키 높은 부품을 먼저 하게 되면, SMD Type 의 부품을 땜하기가 불편하다.
- ⑥ 사용 전원은 9V 300mA 아답터를 사용한다.

* RoHS, WEEE, ELV 란 무엇인가?

유럽 국가(EU)가 선도하여 전 세계적인 흐름이 된 환경 보호 정책의 하나인 전자 제품의 recycling/ re-use 와 유해 물질 사용의 금지는 SMT 분야에도 무연 납땜 공정(Pb-free Soldering Process)이라는 새로운 분야를 이끌어 냈습니다.

2007 년 3 월 1 일부터 중국은 유럽의 RoHS 규정보더 좀 더 엄격한 규정을 실시할 예정입니다.

비단, 전기/전자 분야뿐만이 아니라 모든 산업 분야에서 납 사용 금지 법안을 적용하고 있거나 적용 예정이며, 유럽은 2006 년 7 월 1 일부터 유럽으로 수출되는 모든 전자/전기 제품을 제조자가 회수(Recycling / Re-use)토록 하며, 제조 시 유해 물질(Lead, Carbon, Cadmium, Chromium, Bromine, Mercury or Vinyl Plasticizers)의 사용을 금하고 있습니다. 이에 국내 수출업체를 포함한 모든 전세계 수출기업들은 이미 무연 납땜에 대한 재료, 공정 등을 실시하고 있거나, 2006 년 6 월말까지 실시할 예정입니다.

하기의 내용은 6 가지의 유해 물질의 최대 허용치입니다.





Cadmium (Cd)	:	100 ppm(0.01%)
Mercury (Hg)	:	1000 ppm(0.1%)
Lead (Pb)	:	1000 ppm(0.1%)
Polybrominated Biphenyls (PBB)	:	1000 ppm(0.1%)
Polybrominated Diphenyl Ethers (PBDE)	:	1000 ppm(0.1%)
Hexavalent Chromium (CrVI)	:	1000 ppm(0.1%)

기존의 Sn-Pb 로 구성되어 있는 납땜 재료들은 Pb 가 빠지고 대신 Ag, Cu 나 기타, 다른 물질이 혼합된 새로운 형태의 납땜 재료의 사용을 탄생시켰고, 이에 따른 납땜 공정의 변화로 납땜 공정장비 및 자재의 무연화가 급속히 이뤄지고 있고, 또한 전자 부품, P.C.B, 기타 등등에도 적용되고 있습니다.

전기, 전자분야에 적용되는 EU 법안은 크게 'RoHS (Restriction of the Use of Certain Hazardous Substances in electrical and electronic equipment)'와 'WEEE (Waste Electrical and Electronic Equipment)'의 2 가지로 요약되며, 이 가운데 SMT(PCB Assembly) 분야와 밀접한 관계가 있는 것은 RoHS 입니다.

또한 이미 시행되고 있는 자동차 분야에서의 ELV(End of Life Vehicle)도 폐차 시 RoHS에서 규정하는 유해물질에 대한 규제와 자동차 개발 시 부품의 재사용/리사이클(re-use/recycle)을 권장하고 있습니다.





1.5. KIS-KIT1000L 프로그램 설명 (Study Demo Kit)

1). KIS-R13.56Mhz 프로그램

(1) 프로그램설치

제작한 KIT 를 테스트 하기 위하여 KIS_R13.56MHz.msi 설치한다. 설치가 완료되면 바탕화면에 실행아이콘이 생성됩니다.



Cancel Previous Close





(2) Demo Kit 프로그램을 실행한다.

(3) 전원을 인가하고 Port 를 아래와 같이 설정을 해 준 다음 Connect 를 클릭한다.

VISCOMM 13.56MHz Study DEMO KIT VI.0.0	- 0 🗙
[Communication Control]	
Connect Disconnect	
Port No : Cost I Flow Control	
Pause Resume PORT Boud Rate : 19200 • O None Data Bit : 8 O Xon/Xoff Parity Bit : None O RTS Stop Bit : 1 O Xon/RTS	
Setup the Serial Port Configuration!	
To Excel Delete Ite	ms

a) Port No : Com1 ~Com4

- b) Boud Rate :19200
- c) Data Bit:8
- d) Parity Bit : None
- e) Stop Bit :1

(4) Tag 를 안테나 위에 놓으면 아래와 같이 Tag Data 를 확인할 수 있다.

KISCOMM 13,56MHz Study DEMO MAIN HELP	KIT V1.0.0				
[Communication Control]		[De	tecting Tag List	1	
	CNT	DATE/TIME	TAG NO.	COUNT	
e 🛱	1	11:18:12.059	1152283617000000C	2	
Connect Disconnect	2	11:18:10.276	0606368737000000C	4	
	3	11:18:14.683	1686335969000000C	1	
SEND Pause Resum Difference Pause Resum Difference Pause Pause Difference Pause Pause<					
	Т	Excel		Delete	ltems
	_			_	





2) Hyper Terminal

(1) 환경설정

보조프로그램의 Hyper Terminal 실행하고 아래와 같이 설정한다.

- a) Boud Rate : 19200
- b) Data bit : 8
- c) Parity : n (없음)
- d) Stop bit : 1
- e) Hardware : none(없음)
- (2) 전원인가
- (3) Data 확인

<stx>142366768100000C<etx> -> Tag ID

사용 예)

《ounique - 하이퍼터미널 파일(E) 편집(E) 보기(V) 전화(C) 전송(I) 도움말(H) □☞ ☞ 종 폐금 답	
COMI 등록 정보 ? × 포트 설정 * 최당 비트 수(B): * 데이터 비트(D): 6 패리티(P): 없음 정지 비트(S): 1 호를 컨트롤(E): 없음 고급(A) 기본값 복원(B) 확인 취소 적용(A)	Unique 등록 정보 ? × 연결 대상 설정 ● ● unique ● ● Unique ● · · ● · · ● · · ● · · ● · · ● · · ● · · ● · · ● · · ● · · ● · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · <t< th=""></t<>
연결 0:00:17 자동 감지 9600 8-N-1 SCROLL CAPS NUM 캡	반향

위와 같이 Com Port 설정하고 전원을 인가함.







Tag 을 리더에 가까이 대면 위와 같이 Tag ID 가 PC 로 전송되는 것을 확인할 수 있습니다.





2. 아카데미 키트를 이용한 소프트웨어 제작

2.1. 아카데미 키트와 통신하기

아카데미 키트를 이용한 소프트 웨어 제작을 위해 먼저 통신하는 방법을 알아야 한다. PC와 키트는 시리얼 통신이 가능하기 때문에 시리얼 통신 제어 방법부터 설명 한다.

시리얼 통신을 하기 위해 RS232 케이블로 PC 와 키트를 연결 시키고 소프트웨어 제작을 시작한다. 프로그램 제작 하기 위해 비주얼 베이직 6을 실행시키고 필요한 구성요소 및 참조를 추가한다.

2.1.1. 새 프로젝트 만들기

Visual Studio 6.0 프로그램을 실행 시키고 새 프로젝트를 작성하기 위해 [파일]->새 프로젝트를 선택하고 '새 프로젝트' 대화상자가 나타나면, '표준 EXE'를 더블 클릭한다.

2.1.2. 구성요소 추가

폼을 디자인하기 위해 필요한 구성요소를 추가해야 한다. 구성요소를 추가 하기 위해서는 도구상자 위에서 마우스 오른쪽을 클릭하고 메뉴에서 구성요소를 클릭한다. 또는 상단 메뉴 프로젝트의 하위메뉴에서 구성요소를 클릭한다.

- 시리얼 통신을 하기 위해 Microsoft Comm Control 6.0 추가

- 엑셀 파일로 저장하기 위해 Microsoft Common Dialog Control 6.0 (SP6) 추가

- 플렉스 그리드 사용을 위해 Microsoft FlexGrid Control 6.0 추가

필요한 구성요소 위 3 개를 선택하고 적용을 클릭하면 도구상자에 도구들이 추가된다.

구성요소 추가 화면



2.1.3. 참조 추가

구성요소 추가가 끝나면 참조 파일을 추가하기 위해 상단 메뉴 프로젝트의 하위메뉴 참조를 클릭하면 참조 대화상자가 나타난다.

MDB 를 사용하기 위해 Microsoft ActiveX Data Objects 2.8 Library 추가





- Excel 파일을 만들기 위해 Microsoft Excel 9.0 Object Library 또는 11.0 추가 필요한 참조를 선택하고 적용을 클릭하면 참조 파일이 추가된다.



2.1.4. 시리얼 통신 폼 디자인 및 코드 작성

시리얼 통신을 설정하고 연결하는 Communication Port 프레임은 ComboBox, CommandButton, MSComm 으로 구성되어 있다. ComboBox 는 통신할 포트 번호를 선택하기 위해 필요하고, CommandButton 은 통신 포트를 연결 하고 종료 할 때 필요하며, MSCom 은 시리얼 통신 을 하기 위해 필요하다.

아래의 그림과 표를 참조해서 컨트롤을 삽입하고 속성을 지정한 후 다음과 같은 폼을 디자인 한다.



Comm Open Button 을 클릭하면 Caption 이 'Comm Close'로 변경되고, ComboBox 가 비활성화 되며 다시 클릭하면 복귀된다.

번호	종류	속성
1	Frame	Caption : Communication Port
2	ComboBox	Text : 비워 둠 Enabled : True
3	CommandButton	Caption : Comm Open Style : 1-그래픽
4	MSComm	

폼 디자인이 끝나면 아래의 코드를 입력하여 시리얼 통신을 테스트 한다.

• 먼저 시리얼 통신 연결에 사용할 포트 번호를 선택할 수 있게 콤보박스에 시리얼 번호들을 추가한다.

cboCommPort.Clear

- 콤보박스의 데이터를 지운다.

For i = 1 To 16

- 콤보박스에 COM16 번 까지 데이터를 입력시킨다.



26

Rev.1.0

(주)키스컴 KISCOM	조립키트 매뉴얼
cboCommPort.AddItem "COM" & Str(i) Next I	
cboCommPort.ListIndex = 0	- 콤보박스의 데이터 중 COM1 을 선택한다.
• Communication Port 프레임의 시리얼 통신을	을 설정하고 열고 닫는 코드를 작성한다.
Private Sub cmdPortOpen_Click() On Error GoTo GoErr	
Dim i As Integer	
If cboCommPort.Enabled = True Then strCommBuffer = ""	- 포트 번호 선택 콤보박스가 활성화 상태면 통신을 연결한다.
'// 통신 포트 환경 설정	
With RFIDComm1 .Settings = "38400,N,8,1" .DTREnable = True	- 통신환경 설정
.InBufferCount = 0 .InBufferSize = 2048 .InputLen = 1 .InputMode = comInputModeText	- 수신버퍼 초기화 - 수신버퍼 크기 설정 - 수신버퍼에서 반환할 문자수 설정 - 데이터 수신 모드 설정 - 소시버피 초기회
.OutBufferSize = 2048 .RThreshold = 1 .SThreshold = 0 .RTSEnable = False	- 승진비피 조거와 - 송신버퍼 크기 설정 - comEvReceive 이벤트 핸들러 설정 - comEvSend 이벤트 핸들러 해제 - RTS 회선 해제 => commNone 로 설정할경우.
.ParityReplace = "" .CommPort = cboCommPort.ListIndex .PortOpen = True End With	- 패리티 오류 무시 + 1 - 콤보박스에서 선택한 포트 번호로 연결 - 포트 열기
If RFIDComm1.DSRHolding = True Then cmdPortOpen.Caption = "Comm Close" cboCommPort.Enabled = False	RFIDComm1.RTSEnable = True - "Comm Open" 버튼의 Caption 을 "Comm Close"로 변경한디 - 포트 선택 콤보박스를 비활성화 한다.
Else cmdPortOpen.Caption = "Comm Open" cboCommPort.Enabled = True RFIDComm1.PortOpen = False End If	- 시리얼 통신을 닫을 경우 - "Comm Close" 버튼의 Caption 을 "Comm Open"로 변경한디 - 포트 선택 콤보박스를 활성화한다. - 시리얼 통신을 종료한다.
Exit Sub	- 에러가 없으면 이벤트 프로시저를 종료한다.
GoErr: MsgBox "cmdPortOpen_Click:" & Err.Descri End Sub	ption - 에러가 있으면 메시지 박스로 에러를 보여줌.
 시리얼 통신을 연결하고 RFID 리더기(ACAI 통신으로 보내준다. 이때 시리얼로 들어온 데 	DEMY KIT)에 Tag 를 올리면 리더기에서 태그 ID 를 읽어 시리 [.] 이터를 처리하는 코드를 작성한다.
Private Sub RFIDComm1_OnComm() Dim strRcv As String	
Select Case RFIDComm1.CommEvent Case comEvReceive	- 시리얼 통신으로 데이터를 수신했다면





If RFIDComm1.InBufferCount > 0 Then	
If nMode = MODE_IDLE Then nMode = M strRcv = RFIDComm1.Input	IODE_READ - 수신 데이터를 수신 데이터 버퍼 변수에 저장
Call ParcingData(strRcv)	- 데이터 분석 함수로 이동
End If End Select End Sub	
 수신 데이터의 시작문자와 종료문자 그리고 Tag 작성한다. 	g ID 를 구분하여 오류 없이 데이터가 들어왔는지 분석하는 코드를
Private Const STX As Byte = &H2 Private Const ETX As Byte = &H3	- 시리얼 통신 수신문자의 시작문자 - 시리얼 통신 수신문자의 종료문자
Private Sub ParcingData(ByVal strRcv As String) Dim i As Integer	
Select Case Asc(strRcv) Case ETX 'Call BookSearch(strCommBuffer) - 데이터를 확인하려면 위에 코드를 주석처럼 Case STX strCommBuffer = vbNullString Case Else strCommBuffer = strCommBuffer & strRc End Select End Sub	 수신 문자가 시작문자, 종료문자, Tag ID 인지 구분 종료문자이면 데이터를 처리함 데이터가 다 들어오면 메시지를 확인해 봐야 한다. 리하고 메시지 박스를 띄운다. (MsgBox strCommBuffer) 시작 문자면 버퍼 변수를 초기화 하고 데이터를 저장한다. Tag ID 를 종료문자를 수신할 때 까지 버퍼 변수에 추가한다.



2.2. 파일 저장 및 관리 프로그램 공통 함수

아카데미 키트를 이용한 도서 관리 프로그램과 출석 관리 프로그램을 제작 하기 위해 파일 저장 함수와 관리 프로그램 공통으로 들어가는 함수에 대해 알아본다.

2.2.1. CSV 파일 저장 함수

파일로 저장하기 위해서는 데이터 베이스에서 필요한 정보를 검색해 레코드 셋을 작성한 다음 파일에 저장해야 한다.

• 데이터 베이스의 레코드를 검색하여 레코드 셋에 저장하는 함수를 작성한다.

Public Function SaveCsvFile (strQuery As String, strHead As Stri	ng)
Dim conn As ADODB.Connection	
Dim strFileName As String	- 파일이름을 저장하는 변수
Set conn = New ADODB.Connection	
Set rdSet = New ADODB.Recordset	레시티 께서 두 서 거 귀 다
conn.Open connStr	- 데이터 베이스들 연결한다.
If rdSet.State <> adStateClosed Then	
rdSet.Close	- 레코드 셋이 열려 있으면 닫는다.
End If	
rdSet.Open strQuery, connStr, adOpenKeyset	- 새로운 레코드 셋을 작성한다.
f Lon(ctrFileName) > 0. And rdSet Record(count > 0. Then	- 파일이금을 접역 된아 가져준다. - 파이이르이 기이르 하이하다
Me.MousePointer = vbHourglass	- 마우스 포인터를 모래시계 표시로 변경하다.
Call SaveRdSetToCSVFile(strFileName, strHead, rdSet)	- CSV 파일로 저장하는 프로시저
Me.MousePointer = vbDefault	- 마우스 포인터를 원 상태로 변경한다.
End If	
rdSat Close	- 레코드 세우 다느다
Set conn = Nothing	- 데이터 베이스 연결을 종료하다.
End Function	

• 파일 저장 대화상자를 이용해 파일명을 입력 받아 파일을 만들고 파일명을 받아온다.

Public Function MakeCSVFileName(ByRef dlgfile As Object) As String Dim n, n2 As String Dim strSaveFileName As String On Error GoTo ErrHandle strSaveFileName = "" - 파일 이름을 초기화 With dlgfile .InitDir = "C:₩" - 파일 저장 대화상자의 기본 위치를 지정한다. .DialogTitle = "파일 새로만들기" - 파일 저장 대화상자의 제목을 지정한다. .Filter = "만들파일 (*.csv)" & n & "*.csv" & n - 파일 저장 형식을 지정한다. .FilterIndex = 2 - 파일 저장 기본 확장자를 지정한다. .DefaultExt = "csv" .Flags = cdlOFNAllowMultiselect + cdlOFNExplorer



(주)키스컵 KISCOM	30 조립키트 매뉴얼
.ShowSave .CancelError = True	- 파일 저장 대화상자를 보여준다.
strSaveFileName = .FileName End With	- 파일명을 입력 받아 변수에 저장한다.
If Len(strSaveFileName) < 1 Then MakeCSVFileName = "" Exit Function End If	- 같은 파일이 존재한다면 - Null 값을 반환한다.
MakeCSVFileName = strSaveFileName	- 파일을 생성하고 파일 이름을 반환한다.
Exit Function ErrHandle: MsgBox ("MakeCSVFileName:" & Err.De End Function	escription)
 파일이 만들어 졌으면 레코드 셋에 저장된] 정보를 파일로 옮기는 함수를 작성한다.
Sub SaveRdSetToCSVFile(ByVal strSavel ADODB.Recordset) On Error GoTo GoErr: Dim IFileNum As String Dim ix As Long ' 65000 line 까지 (행값의 Dim jx As Integer '255 까지 (열값의 한계 Dim buffer As String '65 K 까지 (최대 비 Dim jCount As Integer	FileName As String, ByVal strHead As String, ByRef rdSet As 한계)) 퍼 사이즈를 지정함)
lFileNum = FreeFile jCount = (rdSet.Fields.Count) - 1	- 레코드의 길이를 구한다.
Open strSaveFileName For Output As lF buffer = strHead Print #lFileNum, buffer buffer = "" Print #lFileNum, buffer	ileNum - 쓰기 모드로 파일을 오픈 한다. - 파일 제목을 먼저 쓴다.
For ix = 0 To (rdSet.RecordCount - 1) buffer = ""	- 레코드의 길이 만큼 레코드를 파일에 쓴다.
For jx = 0 To jCount buffer = buffer & rdSet(jx) & "," Next jx	- 데이터를 쓰고, 구분자로 콤마를 추가한다.
'마지막 comma 는 삭제 buffer = Left(buffer, Len(buffer) - 1)	
Print #lFileNum, buffer rdSet.MoveNext Next ix	
Close #lFileNum - 저장 상태와 저장 경로 및 이름을 메시지 MagDay "저사러 a 리고자텍스니티 거그며	- 파일에 데이터를 모두 쓰고 파일을 닫는다. 박스로 띄운다.
Misgoox 7878 여프도 시장였급니다. 정도를 Exit Sub GoErr:	- 저장 중 오류가 나면 여기서부터 실행한다.





31

Close #lFileNum	- 파일을 닫는다.
If Err.Description > "" Then MsgBox Err.Description, vbExclamation, "Error !	- Error Message 가 있으면 메시지를 보여준다. !"
End If Fnd Sub	
2.2.2. 데이터 체크 함수	
● 데이터 베이스에 데이터를 저장하기 위해 데이터의 유	r효성을 체크 해야 한다.
Public Function CheckNullData(ByVal strData As Strin Dim i As Integer Dim strTemp As String	ng) As String
strData = Trim(strData) strTemp = ""	- 앞뒤 공백문자를 잘라낸다.
If Trim(strData) = "" Then strTemp = " "	- 변수에 데이터가 없다면 공백문자를 하나 넣는다.
For i = 1 To Len(strData)	- 데이터 길이 만큼 루프하며 데이터를 체크한다.
strTemp = strTemp & "'"	- 데이터에 문자가 (')일 경우 ('')으로 변경한다.
strTemp = strTemp & Mid(strData, i, 1) Next i	
End If	
CheckNullData = strTemp	- 체크가 끝나면 다시 문자를 반환한다.
Exit Function	
GoErr: MagBoy "CheckNullDate:" & Err Description	
End Function	





2.3. 메뉴 작성하기

여러 개의 명령 버튼들을 여기저기 나열하는 것보다는 폼의 맨 위에 메뉴 표시줄을 표시해서 작업을 처리하는 것이 편리하고 보기도 좋기 때문에 등록 창으로 이동하는 것은 메뉴에서 처리한다.

2.3.1. 메뉴 작성 방법

비주얼 베이직에서는 메뉴를 작성할 수 있게 [도구->메뉴 편집기]를 통해 메뉴 편집기 대화상자를 제공한다. 메뉴 편집기 대화상자를 사용하면, 기존의 메뉴 또는 새로운 메뉴를 작성할 수 있다. 메뉴 표시줄도 일반 컨트롤과 마찬가지로 Click 이벤트 프로시저를 통해 작성하면 된다.

• 등록 메뉴를 만들기 위해 메인폼 위에서 마우스 오른쪽을 클릭하면 다음과 같은 메뉴가 나온다.

☑ 코드 보기(<u>0</u>)	
·····································	
■ 전프를 참고가(c) 健 붙여넣기(P)	
☆성(<u>B</u>)	
사용자 정의 컨트롤 업데이트(U)	
메뉴에서 메뉴 편집기를 선택하면 아래와 같은 화면이 나타난다. 도서 관리 프로그램 Ver 1.0.0	
등록	
메뉴 편집기	X
Caption(<u>P</u>): 등록	확인
Name(M): mnuReg	취소
Index(次): Shortcut(S): (없음)	•
HelpContextID(<u>H</u>): 0 NegotiatePosition(<u>O</u>): 0	- 없음 📃
\square Checked(C) \blacksquare Enabled(E) \blacksquare Visible(V) \square	WindowList(<u>W</u>)
◆ ◆ ◆ CER(N) 4°C(1)	삭제(<u>T</u>)
1	

 메뉴 편집기에서 Caption, Name 을 입력하고 삽입버튼을 클릭하면 편집기 아래쪽에 등록되었음을 알 수 있다.
 도서 등록, 회원등록을 하위 메뉴로 하기 위해 오른쪽 방향 화살표를 클릭한 후 화면과 같이 되면 확인을 클릭하여 화면을 닫는다.

Private Sub mnuBookReg_Click()



Rev.1.0



frmBookReg.Show 1 End Sub

Private Sub mnuMemReg_Click() frmMemReg.Show 1 End Sub - 도서 등록 폼을 보여준다.

- 회원 등록 폼을 보여준다.





2.4. 데이터 베이스 구성

도서 관리 프로그램과 출석 관리 프로그램 제작에 필요한 MDB 파일은 아래와 같이 구성되어있다.

2.4.1. 도서 관리 프로그램

도서 관리 프로그램 제작에 필요한 MDB 파일 구성은 도서리스트 테이블과 회원리스트 테이블 그리고 대여/반납 리스트로 구성되어있다.

BOOKLIST 테이블

번호	필드 이름	데이터 형식	필드 크기	비고
1	B_CODE	텍스트	20	도서코드
2	B_TITLE	텍스트	50	도서명
3	B_WRITER	텍스트	50	저자
4	B_PUBLISH	텍스트	50	출판사
5	B_DATE	텍스트	20	발행일
6	B_PRICE	통화		가격
7	B_LEND	텍스트	10	대여/반납 구분
8	B_LENDNUM	숫자	정수(Long)	대여수
9	B_LENDDATE	텍스트	20	대여/반납 일시

MEMBERLIST 테이블

번호	필드 이름	데이터 형식	필드 크기	비고
1	U_SNUM	텍스트	20	학번
2	U_SNAME	텍스트	20	이름
3	U_SSEX	텍스트	10	성별
4	U_SPHON	텍스트	20	연락처
5	U_SEMAIL	텍스트	20	E-Mail
6	U_SADDRESS	텍스트	50	주소

LENDLIST 테이블

번호	필드 이름	데이터 형식	필드 크기	비고
1	LB_ID	일련 번호	정수(Long)	
2	LB_DATE	텍스트	20	대여/반납 날자
3	LB_TIME	텍스트	20	대여/반납 시간
4	LB_BOOKCODE	텍스트	20	도서 코드
5	LB_MEMCODE	텍스트	20	학번
6	LB_LEND	텍스트	5	대여/반납 구분





2.4.2. 출석 관리 프로그램

출석 관리 프로그램 제작에 필요한 MDB 파일 구성은 과목 정보 테이블과 학생 정보 테이블 그리고 출석 현황 리스트 테이블로 구성되어있다.

COURSEINFO 테이블

번호	필드 이름	데이터 형식	필드 크기	비고
1	C_CODE	텍스트	20	과목코드
2	C_NAME	텍스트	20	과목명
3	C_TCHNAME	텍스트	20	담당교수
4	C_STATIME	텍스트	10	출석시간
5	C_ENDTIME	텍스트	10	지각시간

STUDENTINFO 테이블

번호	필드 이름	데이터 형식	필드 크기	비고
1	S_NUM	텍스트	20	학번
2	S_NAME	텍스트	20	이름
3	S_CLASS	텍스트	20	학과
4	S_PHON	텍스트	20	연락처
5	S_EMAIL	텍스트	20	E-Mail

ATTENDANCELIST 테이블

번호	필드 이름	데이터 형식	필드 크기	비고
1	A_ID	일련 번호	정수(Long)	
2	A_DATE	텍스트	10	출석 일자
3	A_TIME	텍스트	10	출석 시간
4	A_STUCODE	텍스트	20	학번
5	A_COUCODE	텍스트	20	과목코드
6	A_STATE	텍스트	5	출석상태





2.5. 도서 관리 프로그램 만들기

도서 관리 프로그램은 리더기에서 Tag 를 읽으면 시리얼 통신을 통해서 Tag ID(도서코드) 가 수신된다. 수신된 도서코드로 데이터 베이스에서 책 정보 테이블을 검색하여 도서코드가 일치하는 책 정보와 현재 시간을 책 정보 프레임에 보여준다. 책 정보가 화면에 나타나면 회원 정보 테이블이 활성화 되면서 회원이름을 선택할 수 있게 되며, 회원을 선택하면 회원 정보가 회원 정보 프레임에 보여진다.

책 정보와 회원 정보가 화면에 나타나면 대여 인지 반납인지 구분하여 버튼을 선택한다. 버튼 클릭 이벤트를 받으면 일자, 시간, 코드, 대여자, 대여여부를 리스트에 나타나고 데이터 베이스 대여 리스트 테이블에 등록된다.

2.5.1. 메인 폼 만들기

프로그램의 구성으로는 도서정보를 등록, 수정, 삭제 할 수 있는 도서등록 화면과 회원 정보를 등록, 수정, 삭제 할 수 있는 회원등록 폼 그리고 메인 화면이 있다.

메인 화면은 다음과 같이 구성되어 있다.

- 책 정보를 보여주는 책 정보 프레임
- 회원을 선택하고 회원 정보를 보여주는 회원 정보 프레임
- 대여와 반납을 선택할 수 있는 대여/반납 구분 프레임
- 시리얼 통신을 설정하고 연결할 수 있는 COMMUNICATION Port 프레임
- 현재 총도서, 소장 도서, 대여 도서의 수를 보여주는 현재 장서 정보 프레임
- 대여 리스트 및 소장 도서 정보를 파일로 저장하는 파일 저장 프레임 그리고 대여 리스트로 구성되어 있다.

■ 도서 관리 프로그램 Ver 1,0,2	X
등록	
_ 책 정보	Communication Port
도서코드 : 3079086340	COM 1 Comm Close
도서명:미실	
저 자:김별아	「현재 장서 정 <mark>보</mark>
출 판 사 : 문이당	총 도서 : 87
발행일: 2005-3-8 가격: 95000	소장 도서 : 80 대여 도서 : 7
대여/반납 일시 : 07-02-01 17:27:01	
	「파일 저장
	대여/반납 리스트 소장 도서 정보
회원선택: 이석호	
<mark>학 변</mark> : H20070005	변호 일 자 시 간 도서명 대여자 대여여부
이 름:이석호 성별:남자	2 07-02-01 17:26:19 살아 있는 동안 꼭 해야 남하원 대여 3 07-02-01 17:26:23 ORACLE ON UNIX 데이 정준선 반납
연락처: 011-235-5656	4 07-02-01 17:26:27 나모 웹에디터 FX 확실 (장서기 반납 5 07-02-01 17:26:32 나모 웹에디터 FX 확실 (당하원 대여
E-Mail : jang22@nate.com	6 07-02-01 17:26:42 ORACLE ON UNIX 데미 김동기 대여 7 07-02-01 17:26:49 살아 있는 동안 꼭 해야 이석호 반납
주 소: 서울시 서초구	8 07-02-01 17:26:54 미실 정준홍 대여
대여/반납 구분	
반 납	

RFID 13.56MHz 스터디킷을 이용한 도서 관리 프로그램




Main Form				
번호	종류	속성		
1	Form	Caption : 도서 관리 프로그램	StartUpPosition : 2 -화면 가운데	

프로그램 메인 화면 로딩 이벤트 프로시저 작성

Private Sub Form_Load() On Error GoTo GoErr:

Call SetEnv	- 폼을 로드하기 전에 기초 설정을 한다.
Call InitGrid	- 플렉스 그리드 초기 화면을 만든다.
Call BookCount	- 총 도서 수, 소장 도서 수, 대여 도서 수를 가져온다.
Call MemberList	- 회원 선택 콤보박스에 추가 할 회원 이름을 가져온다.
Exit Sub	- 에러가 없으면 여기서 프로시저 종료
GoErr:	- 에러 처리
MsgBox "Form_Load:" & Err.Description	- 에러시 메시지 박스로 함수 이름과 에러 메시지를 보여줌
End Sub	- 프로시저 종료

도서관리 메임폼 로드전 콤보박스에 포트번호를 입력시키고 DB 파일과 경로명을 변수에 저장시킨다. •

Private Sub SetEnv() Dim i As Integer

cboCommPort.Clear	- 콤보박스의 데이터를 지운다.	
For i = 1 To 16 cboCommPort.AddItem "COM" & Str(i) Next I	- 콤보박스에 COM16 번 까지 데이터를 입력시킨다.	
cboCommPort.ListIndex = 0	- 콤보박스의 데이터 중 COM1 을 선택한다.	
strDBFile = App.Path & "\db\library"	- DB 파일 경로와 파일명 저장	
connStr = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source = " & strDBFile & ".MDB" connStr = connStr & ";Persist Security Info=False" - MDB 파일 OPEN 설정값을 변수。		

End Sub

2.5.2. 현재 장서 정보 프레임 만들기

현재 소장도서와 대여도서의 수를 보여주는 현재 장서 정보 프레임은 Label 과 Text Box 로 구성되어 있다.

현재 장서 총 도서 :	정보	Frame
소장 도서	0 대여도서 : 0	
Rev.1.0		한국폴리텍 IV



현재 경	상서 정보 프레임은 보여수	-기만 하면 되기 때문에 TexBox 전체 속성 Enabled -> False 로 했다.			
번호	종류	속성			
1	Frame	Caption : 현재 장서 정보			
2	Label	Caption : 총 도서 : BackStyle : 0 -투명			
3	TextBox	Text: 0 Enabled : False			
나머지	Label, TextBox 는 속성	이 동일하다. 			
• L	비아니 베이스 에서 중 도시	이 구, 조경 도시 구, 내여 도시 구들 가져과 와인에 모여군다.			
Public Din Din Din	Function BookCount() n conn As ADODB.Conn n rdSet As ADODB.Recc n strQuery As String	ection ordset			
Set Set	conn = New ADODB.Co rdSet = New ADODB.R	onnection Recordset			
COL	nn.Open connStr	- 데이터 베이스를 연결한다.			
str(- व्हे	Query = "SELECT COUI 등 도서 수를 가져오는 구든	NT(*) FROM BOOKLIST" 문을 작성한다.			
If r 1 Enc	dSet.State <> adStateCl rdSet.Close 1 If	losed Then - 레코드 셋이 열려 있으면 닫는다.			
rdS	et.Open strQuery, conn	Str, adOpenKeyset - 총 도서 수를 가져오는 레코드셋을 작성한다.			
txtl	NBookSum.Text = rdSe	t(0) - 레코드 셋 실행 결과를 총 도서 텍스트 박스에 표출한다.			
rdS	Set.Close	- 레코드 셋을 닫는다.			
str(_ ¢	Query = "SELECT COUI 번에는 대여중인 도서를	NT(*) FROM BOOKLIST WHERE B_LEND = '" & BOOK_LEND1 & "'" 검색하는 구문을 작성한다.			
If r I Enc	dSet.State <> adStateCl rdSet.Close 1 If	losed Then - 레코드 셋이 열려 있으면 닫는다.			
rdS	et.Open strQuery, conn	Str, adOpenKeyset - 대여 도서 수를 가져오는 새로운 레코드 셋을 작성한다.			
txtl	NBookLend.Text = rdSe	et(0) - 레코드 셋 작성 결과를 대여 도서 텍스트 박스에 표출			
rdS	et.Close	- 레코드 셋을 닫는다.			
txtl – ब्	NBookHold.Text = Val() 등 도서 수에서 대여중인 또	txtNBookSum.Text) - Val(txtNBookLend.Text) 도서 수를 빼서 소장 도서 수를 구한다음 텍스트 박스에 표출 한다.			
Set	conn = Nothing	- 데이터 베이스 연결을 종료한다.			
End F	End Function				
2.5.3. 책 정보 프레임 만들기					

• 책 정보를 보여주는 책 정보 프레임은 Label 과 Text Box 로만 구성되어 있다.





번호	종류	속성
1	Frame	Caption : 책 정보
2	Label	Caption : 도서코드 : BackStyle : 0 -투명
3	TextBox	Text : 비워 둠 Enabled : False
	-	

나머지 Label, TextBox 는 속성이 동일하다.

 시리얼 통신으로 종료 문자까지 수신하여 Tag ID 를 모두 수신했다면 Tag ID(도서코드)로 데이터 베이스 책 리스트 테이블에서 해당 도서의 정보를 검색해야 한다.

Private Function BookSearch(bookID As String) Dim conn As ADODB.Connection Dim rdSet As ADODB.Recordset Dim strQuery As String	
Set conn = New ADODB.Connection Set rdSet = New ADODB.Recordset conn.Open connStr	- 데이터 베이스를 연결한다.
strQuery = "SELECT * FROM BOOKLIST WHERE B_(- 데이터 베이스 책 리스트 테이블에서 도서코드로 레코	CODE = '" & bookID & "'" 드를 검색하는 구문을 작성한다.
If rdSet.State <> adStateClosed Then rdSet.Close End If	- 레코드 셋이 열려 있으면 닫는다.
rdSet.Open strQuery, connStr, adOpenKeyset	- 새로운 레코드 셋을 작성한다.
<pre>If rdSet.RecordCount > 0 Then txtBookCode.Text = rdSet("B_CODE") txtTitle.Text = rdSet("B_TITLE") txtWriter.Text = rdSet("B_WRITER") txtPublish.Text = rdSet("B_PUBLISH") txtBDate.Text = rdSet("B_DATE") txtPrice.Text = rdSet("B_PRICE") txtLendDate.Text = Format(Now(), "YY/MM/DD") txtLendTime.Text = Format(Now(), "hh:mm:ss") strLendState = rdSet("B_LEND") frmMemberInfo.Enabled = True frmBookLend.Enabled = True</pre>	 도서코드에 해당하는 레코드를 찾았다면 도서코드를 텍스트 상자에 표출 도서명을 텍스트 상자에 표출 저자를 텍스트 상자에 표출 출판사를 텍스트 상자에 표출 발행일을 텍스트 상자에 표출 가격을 텍스트 상자에 표출 대여 또는 반납 날자를 현재 날자로 표출 대여 또는 반납 시간을 현재 시간으로 표출 현재 대여 상태를 가져와 변수에 저장 회원 정보를 선택 할 수 있는 프레임을 활성화 대여/반납 프레임을 활성화



조립 키트 매뉴얼 40
- 해당 도서를 검색하지 못하였다면 메시지박스를 띄움
- 레코드 셋을 닫는다. - 데이터 베이스 연결을 종료한다.

2.5.4. 회원 정보 프레임 만들기

• 회원 정보를 보여주는 회원 정보 프레임은 Label , Text Box, ComboBox 로 구성되어 있다.

_회원 정보		 Frame
회원선택		 ComboDor
학 번:		Сошовох
이 름:	성별 :	
연락처፡		TextBox
E-Mail :		
주 소:		Label

회원 정보 프레임은 비활성화 상태에서 책정보가 보여지면 활성화 상태로 변경된다.

회원 정보 프레임은 보여주기만 하면 되기 때문에 TexBox 전체 속성 Enabled -> False 로 했다.

번호	が示	속성	
1	Frame	Caption : 회원 정보 Enabled : False	
2	ComboBox	Text : 비워 둠 Enabled : True	
3	Label	Caption : 회원선택 : BackStyle : 0 -투명	
4	TextBox	Text : 비워 둠 Enabled : False	
1 1 1			

나머지 Label, TextBox 는 속성이 동일하다.

• 데이터 베이스에서 회원을 검색하여 회원 정보 프레임의 회원선택 콤보박스 리스트를 작성한다.

Private Function MemberList()

```
Dim conn As ADODB.Connection
Dim rdSet As ADODB.Recordset
Dim strQuery As String
```

Set conn = New ADODB.Connection Set rdSet = New ADODB.Recordset conn.Open connStr

- 데이터 베이스를 연결한다.

strQuery = "SELECT * FROM MEMBERLIST ORDER BY U_SNAME ASC" - 데이터 베이스 회원리스트 테이블에서 총 회원 정보를 가져오는 구문을 작성한다.





If rdSet.State <> adStateClosed Then rdSet.Close - 레코드 셋이 열려 있으면 닫는다. End If rdSet.Open strQuery, connStr, adOpenKeyset - 새로운 레코드 셋을 작성한다. - 회원선택 콤보박스의 데이터를 모두 지운다. cboMember.Clear If rdSet.RecordCount > 0 Then - 회원이 있을 경우 콤보박스 리스트에 추가한다. Do While Not rdSet.EOF - 레코드 셋의 마지막 레코드까지 루프한다. cboMember.AddItem Trim(rdSet("U_SNAME")) - 회원이름을 콤보박스에 추가한다. rdSet.MoveNext - 다음 레코드로 이동한다. Loop - 콤보박스에 추가된 회원들 중 첫번째 회원을 선택한다. cboMember.ListIndex = 0 End If - 레코드 셋을 닫는다. rdSet.Close Set conn = Nothing - 데이터 베이스 연결을 종료한다. End Function 데이터 베이스에서 도서 코드로 책 정보를 검색하였다면 대여 또는 반납하는 회원 정보를 선택해야 한다. Private Sub cboMember_Click() If frmMemberInfo.Enabled = True Then - 회원 정보를 선택할 수 있는 상태라면 Call MemberSearch(cboMember.Text) - 선택한 회원의 이름으로 데이터 베이스 회원 리스트 테이블의 해당 회원 정보를 가져온다. End If End Sub 회원 선택 콤보박스에서 회원을 선택하였다면 데이터 베이스 회원 리스트 테이블에서 해당 회원의 정보를 검색해야 한다. Private Function MemberSearch(strMemName As String) Dim conn As ADODB.Connection Dim rdSet As ADODB.Recordset Dim strQuery As String Set conn = New ADODB.Connection Set rdSet = New ADODB.Recordset - 데이터 베이스를 연결한다. conn.Open connStr strQuery = "SELECT * FROM MEMBERLIST WHERE U_SNAME = '" & strMemName & "'" - 데이터 베이스 회원 리스트 테이블에서 회원이름으로 레코드를 검색하는 구문을 작성한다. If rdSet.State <> adStateClosed Then - 레코드 셋이 열려 있으면 닫는다. rdSet.Close End If - 새로운 레코드 셋을 작성한다. rdSet.Open strQuery, connStr, adOpenKeyset If rdSet.RecordCount > 0 Then - 학번을 텍스트 상자에 표출 txtScode.Text = rdSet("U_SNUM") txtSname.Text = rdSet("U_SNAME") - 이름을 텍스트 상자에 표출 - 성별을 텍스트 상자에 표출 txtSsex.Text = rdSet("U_SSEX") - 연락처를 텍스트 상자에 표출 txtSphon.Text = rdSet("U_SPHON") txtSemail.Text = rdSet("U_SEMAIL") - E-Mail 을 텍스트 상자에 표출 txtSaddr.Text = rdSet("U_SADDRESS") - 연락처를 텍스트 상자에 표출

조립 키트 매뉴얼





2.5.5. 대여/반납 구분 프레임 만들기

• 대여, 반납을 선택할수 있는 대여/반납 프레임은 CommandButton 으로 구성되어 있다.



대여/반납 구분 프레임은 비활성화 상태에서 책정보가 보여지면 활성화 상태가 된다.

번호	多 市	속성
1	Frame	Caption : 대여/반납 구분 Enabled : False
2	CommandButton	Caption : 대여 Style : 1-그래픽

 책 정보와 회원 정보가 모두 입력 되었다면 책 정보의 대여 상태를 검색하여 대여중이면 버튼의 글자를 반납으로 바꾸고 반납된 상태라면 버튼의 글자를 대여로 바꾼다. 현재 상태의 대여 또는 반납 버튼을 클릭하면 대여/반납 리스트에 저장해야 한다.

```
- 도서를 대여 또는 반납 버튼을 클릭했다면
Private Sub cmdBookLend_Click()
  On Error GoTo GoErr:
  If txtScode.Text = vbNullString Then
                                           - 회원 정보를 선택하지 않았거나 회원이 아니라면
    MsgBox "회원 정보를 입력하세요"
                                           - 회원 정보를 입력하라는 메시지 박스를 띄우고
    Exit Sub
                                           - 프로시저를 종료한다.
  Else
    If Trim(cmdBookLend.Caption) = "대 여" Then
                                          - 현재 번튼이 대여 버튼이면
      Call AddLendList(BOOK LEND1)
      - 대여/반납 구분을 대여로 하여 대여/반납 리스트에 대여 정보를 추가한다.
                                           - 현재 번튼이 반납 버튼이면
    Else
      Call AddLendList(BOOK_LEND2)
      - 대여/반납 구분을 반납으로 하여 대여/반납 리스트에 반납 정보를 추가한다.
    End If
    frmMemberInfo.Enabled = False
                                           - 회원 정보 프레임을 비활성화 한다.
    frmBookLend.Enabled = False
                                           - 대여/반납 구분 프레임을 비활성화 한다.
                                           - 현재 장서 정보를 재 검색한다.
    Call BookCount
    Call InitInfo
                                           - 책 정보 및 회원 정보를 모두 지운다.
  End If
  Exit Sub
GoErr:
  MsgBox "CmdBookLend_Click:" & Err.Description
End Sub
```





43

조립 키트 매뉴얼 대여 또는 반납했을 경우 대여/반납 리스트를 데이터 베이스 및 플렉스 그리드에 추가해야 한다. Private Function AddLendList(strLend As String) Dim conn As ADODB.Connection Dim strQuery As String Dim strBCode As String, strDate As String, strTime As String, strBTitle As String, strBPublish As String, strBName As String, strSnum As String Dim i As Integer Set conn = New ADODB.Connection - 데이터 베이스를 연결한다. conn.Open connStr - 도서 코드를 변수에 저장한다. strBCode = txtBookCode.Text - 도서명을 변수에 저장한다. strBTitle = txtTitle.Text strBName = txtWriter.Text - 저자를 변수에 저장한다. - 출판사를 변수에 저장한다. strBPublish = txtPublish.Text strDate = txtLendDate.Text - 대여/반납 일자를 변수에 저장한다. strTime = txtLendTime.Text - 대여/반납 시간을 변수에 저장한다. - 학번을 변수에 저장한다. strSnum = txtScode.Text With MSFlexGrid1 - 대여/반납 리스트에 정보를 추가한다. .Rows = .Rows + 1.TextMatrix(.Rows - 1, 0) = .Rows - 1 - 리스트 번호를 리스트에 추가한다. - 대여/반납 일자를 리스트에 추가한다. .TextMatrix(.Rows - 1, 1) = strDate .TextMatrix(.Rows - 1, 2) = strTime - 대여/반납 시간을 리스트에 추가한다. .TextMatrix(.Rows - 1, 3) = strBCode - 도서 코드를 리스트에 추가한다. - 학번을 리스트에 추가한다. .TextMatrix(.Rows - 1, 4) = strSnum .TextMatrix(.Rows - 1, 5) = strLend - 대여 또는 반납을 리스트에 추가한다. End With strQuery = "INSERT INTO LENDLIST(LB_DATE, LB_TIME, LB_BOOKCODE, LB_MEMCODE, LB_LEND) VALUES (" & CheckNullData(strDate) & "', " & _ "'" & CheckNullData(strTime) & "', " & _ "'" & CheckNullData(strBCode) & "', " & _ "'" & CheckNullData(strSnum) & "', " & _ "'" & CheckNullData(strLend) & "')" - 데이터 베이스 대여 리스트 테이블에 대여정보를 추가하는 구문을 작성한다. - 대여정보 레코드를 테이블에 추가한다. conn.Execute strQuery strQuery = "UPDATE BOOKLIST SET B_LEND = '" & CheckNullData(strLend) & "', " & . " B_LENDDATE = '" & CheckNullData(strDate & " " & strTime) & "'" & _ "WHERE B_CODE = " & CheckNullData(strBCode) & "" - 데이터 베이스 도서정보 테이블에 대여상태와 대여시간을 변경하는 구문을 작성한다. - 해당 레코드의 대여상태와 대여시간을 변경한다. conn.Execute strQuery - 데이터 베이스 연결을 종료한다. Set conn = Nothing End Function 2.5.6. 파일 저장 프레임 및 대여 정보 리스트 만들기

대여/반납 리스트와 소장 도서 정보를 파일로 저장할 수 있는 파일 저장 프레임은 CommonDialog , CommandButton 으로 구성되어 있다. 그리고 아래 대여/반납 리스트를 보여주는 MSFlexGrid 를 배치 하였다.

한국폴리텍 IV



44

<mark>파일 저장</mark> 대여/반납 리스트	소장 도서 정보	Frame CommonDialog
		CommandButton MSFlexGrid

번호	종류	속성
1	Frame	Caption : 파일 저장
2	CommandButton	Caption : 대여/반납 리스트 Style : 1-그래픽
3	CommandButton	Caption : 소장 도서 정보 Style : 1-그래픽
4	CommonDialog	
5	MSFlexGrid	

• 대여/반납 리스트와 소장 도서 정보를 저장하기 위한 파일 저장 프로시저를 작성한다.

Private Sub cmdBookInfoSave_Click() - 소장 도서 정보를 파일로 저장한다. Dim strQuery As String, strHead As String strQuery = "SELECT * FROM BOOKLIST ORDER BY B_TITLE ASC" - 데이터 베이스 책 리스트 테이블의 모든 도서 정보 레코드를 검색하는 구문을 작성한다. strHead = cmdBookInfoSave.Caption - 파일 제목을 버튼 Caption 으로 지정한다. Call BookInfoSave(strQuery, strHead) - 도서 정보를 파일로 저장한다. Call BookInfoExcelSave(strQuery, strHead) - 도서 정보를 엑셀 파일로 저장한다. End Sub Private Sub cmdLendListSave_Click() - 대여/반납 리스트 정보를 파일로 저장한다. Dim strQuery As String, strHead As String strQuery = "SELECT LENDLIST.LB_DATE, LENDLIST.LB_TIME, LENDLIST.LB_BOOKCODE, LENDLIST.LB_MEMCODE, LENDLIST.LB_LEND, " & _ "BOOKLIST.B_TITLE, BOOKLIST.B_WRITER, BOOKLIST.B_PUBLISH, " & _ "MEMBERLIST.U_SNAME " & _ "FROM (LENDLIST LEFT JOIN BOOKLIST ON LENDLIST.LB_BOOKCODE = BOOKLIST.B_CODE) " & _ "LEFT JOIN MEMBERLIST ON LENDLIST.LB_MEMCODE = MEMBERLIST.U_SNUM" - 데이터 베이스 대여 리스트 테이블의 대여 정보 레코드로 회원 리스트 테이블과 도서 정보 리스트 테이블의 모든 데이터를 검색하는 구문을 작성한다. 한국폴리텍 IV



strHead = cmdLendListSave.Caption Call BookInfoSave(strQuery, strHead) Call BookInfoExcelSave(strQuery, strHead) End Sub - 파일 제목을 버튼 Caption 으로 지정한다. - 도서 정보를 파일로 저장한다.

- 도서 정보를 엑셀 파일로 저장한다.

• MSFlexGrid 크기를 지정하고 필드명을 입력하여 초기화 한다.

```
Private Sub InitGrid()
  With MSFlexGrid1
    .Clear
    .Rows = 1
    .Cols = 6
    .FormatString = ">번호|^일 자|^시 간|^코 드|^대여자|^대여여부"
    .ColWidth(0) = 450
                                         - 번호 필드의 크기를 지정함
    .ColWidth(1) = 1000
                                         - 대여일자
                                         - 대여시간
    .ColWidth(2) = 1000
                                         - 코드
    .ColWidth(3) = 2000
                                         - 대여자
    .ColWidth(4) = 1440
                                         - 대여여부
    .ColWidth(5) = 800
  End With
End Sub
```





2.5.7. 도서 등록 폼 만들기

도서 등록 화면은 다음과 같이 구성되어 있다.

- 도서 정보를 보여주는 프레임
- 찾고자 하는 도서를 검색할 수 있는 검색 프레임
- 소장 도서 목록을 보여주고 검색 결과를 보여주는 리스트로 구성되어 있다.
- 도서 등록 화면은 메인에서 메뉴를 선택하면 나타난다. 도서 등록 화면에서 필요한 기능은 DB 에서 등록된 도서들을 리스트에 보여주고 도서를 등록, 수정, 삭제 할 수 있는 기능과 도서명, 저자, 출판사로 도서를 검색 할 수 있는 기능이 필요하다.



번호	종류	속성
1	Frame	Caption : 비워 둠
1	Frame	Caption : 검색
2	Label	Caption : 도서코드 : BackStyle : 0 -투명
3	TextBox	Text : 비워 둠
4	CommandButton	Caption : 등록/수정 Style : 1-그래픽
6	MSFlexGrid	

나머지 Label, TextBox, CommandButton 은 속성이 동일하다.



(주)키스컴 KISCOM 조립키	기트 매뉴얼
 도서 등록 폼이 실행되면 도서 목록전체를 검색하기 	위해 데이터 베이스 위치 및 파일명을 지정한다.
Private Sub Form_Load() Dim strQuery As String	
strDBFile = App.Path & "\db\library" connStr = "Provider=Microsoft.Jet.OLEDB.4.0;Data connStr = connStr & ";Persist Security Info=False" Call InitGrid	- DB 파일 경로와 파일명을 변수에 저장한다. Source = " & strDBFile & ".MDB" - DB 파일 OPEN 설정 값을 변수에 저장한다. - 그리드를 초기화 한다.
strQuery = "SELECT B_CODE, B_TITLE, B_WRITE "FROM BOOKLIST ORDER BY B_COD - 데이터 베이스 도서정보 테이블의 도서 정보를 검색 Call SearchData(strQuery)	R, B_PUBLISH, B_DATE, B_PRICE " & _ E" 하는 구문을 작성한다. - 도서 정보를 검색하는 함수를 실행한다.
End Sub	
 도서 목록을 리스트에 보여주기 위해 먼저 리스트의 - 	필드 수와 필드 명을 입력한다.
Private Sub InitGrid() With MSFlexGrid1 .Clear	- 플렉스 그리드를 모두 지운다.
.Rows = 1	
.0015 - 7	- 필드 이름과 크기를 정한다.
.FormatString = ">번호 ^도서코드 <도서명 <저지 .ColWidth(0) = 450 '번호 .ColWidth(1) = 1800 '도서코드 .ColWidth(2) = 2700 '도서명 .ColWidth(3) = 500 '도서명 .ColWidth(4) = 500 '출판사 .ColWidth(5) = 500 '발행일 .ColWidth(6) = 920 '가격 End With End Sub	⊦ <줄판사 <발행일 <가격"
 데이터 베이스에서 도서정보 테이블의 전체 레코드를 	· 읽어와 플렉스 그리드로 보여준다.
Private Function SearchData(strQuery As String) Dim conn As ADODB.Connection Dim rdSet As ADODB.Recordset Dim nCnt, nRtn As Integer	
Call InitGrid	- 플렉스 그리드를 초기화 한다.
Set conn = New ADODB.Connection Set rdSet = New ADODB.Recordset conn.Open connStr	- 데이터 베이스를 연결한다.
rdSet.Open strQuery, connStr, adOpenKeyset	- 새로운 레코드 셋을 작성한다.
If rdSet.RecordCount > 0 Then While Not rdSet.EOF With MSFlexGrid1 .Rows = Rows + 1	- 레코드 셋에 레코드가 있다면 - 레코드 수 만큼 데이터를 플렉스 그리드에 추가한다.
.TextMatrix(.Rows - 1, 0) = .Rows - 1	- 리스트 번호를 리스트에 추가한다.



.TextMatrix(.Rows - 1, 1) = rdSet("B_CODI	군") - 도서 코드를 리스트에 추가한다.
.TextMatrix(.Rows - 1, 2) = rdSet("B_TITL	正") - 도서명을 리스트에 추가한다.
.TextMatrix(.Rows - 1, 3) = rdSet("B_WRIT	`ER") - 저자를 리스트에 추가한다.
.TextMatrix(.Rows - 1, 4) = rdSet("B_PUBI	_ISH") - 출판사를 리스트에 추가한다.
.TextMatrix(.Rows - 1, 5) = rdSet("B_DATI	E") - 발행일을 리스트에 추가한다.
.TextMatrix(.Rows - 1, 6) = rdSet("B PRIC	E") - 가격을 리스트에 추가하다.
End With	
rdSet MoveNext	- 다음 레코드로 이동하다
Wend	
Fleo	
MagDay "드로디 데이터가 어스니다"	- 거섀티 레그드기 어디며 메시기 바ㅅ르 띠ㅇ디
MISEDUX 등득관 데이디가 값급니다 Data M	- 섬색관 데고드가 없다던 메시지 릭스물 ㅋ군다.
	케ㅋㄷ 레이 다느니
rdSet.Close	- 데고드 셋술 딛근다. 레이티 께서 2 성격이 조그하다
Set conn = Nothing	- 데이터 베이스 연결을 중묘안다.
and Function	
 데이터 베이스 도서정보 테이블에 도서를 등록, 수정 ; 	하기 위해 도서 정보를 입력하고 등록 또는 수정한다.
Private Sub cmdReg_Click()	
Dim conn As ADODB.Connection	
Dim rdSet As ADODB.Recordset	
Dim nCnt, nRtn As Integer	
Dim strQuery As String	
If Not CheckValidate Then Exit Sub	- 필수 입력 데이터가 입력되지 않았는지 확인한다.
Set conn = New ADODB.Connection	
Set rdSet = New ADODB.Recordset	
conn.Open connStr	- 데이터 베이스를 연결한다.
strQuery = "SELECT * FROM BOOKLIST WHERE B - 데이터 베이스 도서정보 테이블에 같은 도서코드를 기	5_CODE = '" & Trim(txtBookCode.Text) & "'" 사진 레코드가 있는지 검색하는 구문을 작성하다
If rdSet.State <> adStateClosed Then	
rdSet.Close	- 레코드 셋이 열려 있으면 닫는다.
End If	
rdSet.Open strQuery, connStr, adOpenKeyset	- 새로운 레코드 셋을 작성한다.
If rdSet.BOF Or rdSet.EOF Then	- 같은 도서 코드가 없다면
- 메시지 박스로 등록 여부를 물어보고, OK 을 클릭하'	면 등록 구문을 실행하고, Cancel을 클릭하면 등록하지 :
프로시저를 종료한다.	
nRtn = MsgBox("데이터를 등록하시겠습니까", vbO K nPtn (A sch OK Then	KCancel + vbInformation, "확인")
II III(II \/ VDUK THEII	
EXIL SUD	
strQuery = "INSERT INTO BOOKLIST VALUES (" & CheckNullData(txtBookCode.Text) & "', " & _
"'" & CheckNullData(tx	tBookTitle.Text) & "', " & _
"'" & CheckNullData(tx	tBookWriter.Text) & "', " & _
"'" & CheckNullData(tx	tBookPublish.Text) & "', " & _
"'" & CheckNullData(tx	tBookDate.Text) & "', " & _
"'" & CheckNullData(tx	tBookPrice.Text) & "', " & _
"'" & CheckNullData(BC)OK_LEND2) & "', '0', " & _
"'" & CheckNullData(Fc	ormat(Now(), "YY/MM/DD") & _
" " & Format(Now(), "h	h:mm:ss")) & "')"
- 데이터 베이스 도서정보 테이블에 도서정보를 추	가하는 구문을 작성한다.
Flee	
F126	





- 같은 도서 코드가 있다면 수정해야 할 것이다. 수정 여부를 확인한다. nRtn = MsgBox("데이터를 수정하시겠습니까", vbOKCancel + vbInformation, "확인") If nRtn <> vbOK Then Exit Sub End If strQuery = "UPDATE BOOKLIST " & _ "SET B_TITLE = " & CheckNullData(txtBookTitle.Text) & "', " & _ B_WRITER = '" & CheckNullData(txtBookWriter.Text) & "', " & _ B_PUBLISH = " & CheckNullData(txtBookPublish.Text) & ", " & _ B_DATE = '" & CheckNullData(txtBookDate.Text) & "', " & _ B_PRICE = "" & CheckNullData(txtBookPrice.Text) & """ & _ " WHERE B_CODE = " & CheckNullData(txtBookCode.Text) & "" - 데이터 베이스 도서정보 테이블에 도서정보를 수정하는 구문을 작성한다. End If conn.Execute strQuery, nCnt - 도서정보 레코드를 테이블에 추가 또는 수정한다. Set conn = Nothing - 데이터 베이스 연결을 종료한다. strQuery = "SELECT B_CODE, B_TITLE, B_WRITER, B_PUBLISH, B_DATE, B_PRICE " & _ "FROM BOOKLIST ORDER BY B_CODE" - 데이터 베이스 도서정보 테이블의 도서 정보를 재 검색하는 구문을 작성한다. Call SearchData(strQuery) - 도서 정보를 검색하는 함수를 실행한다. End Sub 데이터 베이스 도서정보 테이블에 도서를 삭제 하기 위해 도서 코드를 입력하고 삭제한다. Private Sub cmdDelete_Click() Dim conn As ADODB.Connection Dim nCnt, nRtn As Integer Dim strQuery As String - 메시지 박스로 삭제 여부를 물어보고, OK을 클릭하면 삭제 구문을 실행하고, Cancel을 클릭하면 삭제하지 않고 프로시저를 종료한다. nRtn = MsgBox("데이터를 삭제하시겠습니까", vbOKCancel + vbInformation, "확인") If nRtn <> vbOK Then Exit Sub End If Set conn = New ADODB.Connection conn.Open connStr strQuery = "DELETE FROM BOOKLIST WHERE B_CODE = '" & CheckNullData(txtBookCode.Text) & "'" - 데이터 베이스 도서정보 테이블에서 도서코드와 일치하는 도서 정보를 삭제하는 구문을 작성한다. conn.Execute strQuery, nCnt - 해당 도서정보 레코드를 삭제한다. - 데이터 베이스 연결을 종료한다. Set conn = Nothing strQuery = "SELECT B_CODE, B_TITLE, B_WRITER, B_PUBLISH, B_DATE, B_PRICE " & _ "FROM BOOKLIST ORDER BY B_CODE" - 데이터 베이스 도서정보 테이블의 도서 정보를 재 검색하는 구문을 작성한다. Call SearchData(strQuery) - 도서 정보를 검색하는 함수를 실행한다. End Sub

한국폴리텍 IV



도서정보 테이블에서 검색하고자 하는 도서정보를 도서명, 저자, 출판사로 검색할 수 있는 기능을 추가한다. Private Sub cmdSearch_Click() Dim strQuery As String Dim intNum As Integer intNum = 0strQuery = "SELECT B_CODE, B_TITLE, B_WRITER, B_PUBLISH, B_DATE, B_PRICE FROM BOOKLIST " - 데이터 베이스 도서정보 테이블의 도서 정보를 재 검색하는 구문을 작성한다. If txtSBookTitle.Text <> "" Then - 도서명으로 검색하고자 하면 구문을 추가 작성한다. If intNum = 0 Then strQuery = strQuery & "WHERE B_TITLE Like '" & txtSBookTitle.Text & "%' " End If intNum = intNum + 1End If If txtSBookWriter.Text <> "" Then If intNum = 0 Then - 저자로 검색하고자 하면 구문을 추가 작성한다. strQuery = strQuery & "WHERE B_WRITER Like '" & txtSBookWriter.Text & "%' " - 앞 구문에 조건문이 있을 경우 추가 조건문을 작성한다. Else strQuery = strQuery & "And B_WRITER Like '" & txtSBookWriter.Text & "%' " End If intNum = intNum + 1 End If If txtSBookPublish.Text <> "" Then If intNum = 0 Then - 출판사로 검색하고자 하면 구문을 추가 작성한다. strQuery = strQuery & "WHERE B_PUBLISH Like '" & txtSBookPublish.Text & "%' " - 앞 구문에 조건문이 있을 경우 추가 조건문을 작성한다. Else strQuery = strQuery & "And B_PUBLISH Like '" & txtSBookPublish.Text & "%' " End If intNum = intNum + 1End If strQuery = strQuery & "ORDER BY B_CODE" - 도서 코드순으로 정렬시킨다. Call SearchData(strQuery) - 도서 정보를 검색하는 함수를 실행한다. End Sub 도서명, 저자, 출판사 입력 텍스트 박스에서 엔터키를 눌러 검색할 수 있는 기능을 추가한다. Private Sub txtSBookTitle_KeyDown(KeyCode As Integer, Shift As Integer) If KeyCode = 13 Then Call cmdSearch_Click End If End Sub 데이터의 길이로 필수 텍스트 상자에 입력 여부를 알아보고 입력되지 않은 곳에 포커스를 주고 종료한다. Private Function CheckValidate() As Boolean If Len(txtBookCode.Text) = 0 Then txtBookCode.SetFocus MsgBox "도서코드를 입력하세요" CheckValidate = False





Exit Function End If

If Len(txtBookTitle.Text) = 0 Then txtBookTitle.SetFocus MsgBox "도서명을 입력하세요" CheckValidate = False Exit Function End If End Function





2.5.8. 회원 등록 폼 만들기

회원 등록 화면은 다음과 같이 구성되어 있다.

- 회원 정보를 보여주는 프레임
- 찾고자 하는 회원을 검색할 수 있는 검색 프레임
- 회원 목록을 보여주고 검색 결과를 보여주는 리스트로 구성되어 있다.
- 회원 등록 화면은 메인에서 메뉴를 선택하면 나타난다. 회원 등록 화면에서 필요한 기능은 DB에 등록된 회원들을 리스트에 보여주고 회원을 등록, 수정, 삭제 할 수 있는 기능과 학번, 이름으로 회원을 검색 할 수 있는 기능이 필요하다.



번호	종류	속성
1	Frame	Caption : 비워 둠
1	Frame	Caption : 검색
2	Label	Caption : 학번 : BackStyle : 0 -투명
3	TextBox	Text : 비워 둠
4	CommandButton	Caption : 등록/수정 Style : 1-그래픽
6	MSFlexGrid	
1-1-1		ID · · · · · · · · · · · · · · · · · · ·

나머지 Label, TextBox, CommandButton 은 속성이 동일하다.

• 회원 등록 폼이 실행되면 회원 목록전체를 검색하기 위해 데이터 베이스 위치 및 파일명을 지정한다.



한국폴리텍 IV

청주대학



Private Sub Form_Load() Dim strQuery As String	
strDBFile = App.Path & "WdbWlibrary" connStr = "Provider=Microsoft.Jet.OLEDB.4.0;Data connStr = connStr & ";Persist Security Info=False" Call InitGrid	- DB 파일 경로와 파일명을 변수에 저장한다. Source = " & strDBFile & ".MDB" - DB 파일 OPEN 설정 값을 변수에 저장한다. - 그리드를 초기화 한다.
strQuery = "SELECT U_SNUM, U_SNAME, U_SSE "FROM MEMBERLIST ORDER BY U_SNUM	X, U_SPHON, U_SEMAIL, U_SADDRESS " & _ //"
- 데이터 베이스 회원 정보 테이블의 회원 정보를 검색	하는 구분을 작성한다.
Call SearchData(strQuery)	- 회원 정보를 검색한다.
End Sub	
 회원 목록을 리스트에 보여주기 위해 먼저 리스트의 	필드 수와 필드 명을 입력한다.
Private Sub InitGrid()	
With MSFlexGrid1 .Clear .Rows = 1	- 플렉스 그리드를 모두 지운다.
.Cols = 7	- 푀ㄷ 이르과 ㅋ기르 저하다
 .FormatString = ">번호 ^학 번 ^이 름 ^성별 < .ColWidth(0) = 450 '번호 .ColWidth(1) = 1000 '학번 .ColWidth(2) = 800 '이름 .ColWidth(3) = 620 '성별 .ColWidth(4) = 1400 '연락처 .ColWidth(5) = 1100 'E-Mail .ColWidth(6) = 2000 '주소 End With End Sub 데이터 베이스에서 회원 정보 테이블 전체를 읽어와 Private Function SearchData(strQuery As String) 	연락처 <e-mail <주 소"<br="">플렉스 그리드로 보여준다.</e-mail <주>
On Error GoTo GoErr:	
Dim conn As ADODB.Connection Dim rdSet As ADODB.Recordset Dim nCnt, nRtn As Integer	
Call InitGrid	- 플렉스 그리드를 초기화 한다.
Set conn = New ADODB.Connection	
Set rdSet = New ADODB.Recordset conn.Open connStr	- 데이터 베이스를 연결한다.
rdSet.Open strQuery, connStr, adOpenKeyset	- 새로운 레코드 셋을 작성한다.
If rdSet.RecordCount > 0 Then While Not rdSet.EOF With MSFlexGrid1 .Rows = .Rows + 1	- 레코드 셋에 레코드가 있다면 - 레코드 수 만큼 데이터를 플렉스 그리드에 추가한다.





.TextMatrix(.Rows - 1, 0) = .Rows - 1 .TextMatrix(.Rows - 1, 1) = rdSet("U_SNUM") .TextMatrix(.Rows - 1, 2) = rdSet("U_SNAME" .TextMatrix(.Rows - 1, 3) = rdSet("U_SSEX") .TextMatrix(.Rows - 1, 4) = rdSet("U_SPHON" .TextMatrix(.Rows - 1, 5) = rdSet("U_SEMAIL .TextMatrix(.Rows - 1, 6) = rdSet("U_SADDRE End With rdSet.MoveNext Wend Else	 리스트 번호를 리스트에 추가한다. 학번을 리스트에 추가한다. 이름을 리스트에 추가한다. 성별을 리스트에 추가한다. 성별을 리스트에 추가한다. 연락처를 리스트에 추가한다. E-Mail 을 리스트에 추가한다. 주소를 리스트에 추가한다. 다음 레코드로 이동한다.
MsgBox "등록된 데이터가 없습니다" End If	- 검색된 레코드가 없다면 메시지 박스를 띄운다.
rdSet.Close Set.com = Nothing	- 레코드 셋을 닫는다. - 데이터 베이스 여겨으 조르하다
End Function	에이디 메이프 근실할 승과한다.
• 데이터 베이스 회원 정보 테이블에 회원을 등록, 수정 하기	이 위해 회원 정보를 입력하고 등록 또는 수정한다.
Dim conn As ADODB.Connection Dim rdSet As ADODB.Recordset Dim nCnt, nRtn As Integer Dim strQuery As String	
If Not CheckValidate Then Exit Sub	- 필수 입력 데이터가 입력되지 않았는지 확인한다.
Set conn = New ADODB.Connection Set rdSet = New ADODB.Recordset conn.Open connStr	- 데이터 베이스를 연결한다.
strQuery = "SELECT * FROM MEMBERLIST WHERE U - 데이터 베이스 회원 정보 테이블에 같은 학번을 가진 레코	L_SNUM = ''' & Trim(txtSnum.Text) & "'" ユ드가 있는지 검색하는 구문을 작성한다.
If rdSet.State <> adStateClosed Then rdSet.Close End If	- 레코드 셋이 열려 있으면 닫는다.
rdSet.Open strQuery, connStr, adOpenKeyset	- 새로운 레코드 셋을 작성한다.
If rdSet.BOF Or rdSet.EOF Then - 메시지 박스로 등록 여부를 물어보고, OK 을 클릭하면 등 프로시저를 종료한다.	- 같은 도서 코드가 없다면 등록 구문을 실행하고, Cancel 을 클릭하면 등록하지 않고
nRtn = MsgBox("데이터를 등록하시겠습니까", vbOKCa If nRtn <> vbOK Then Exit Sub	ancel + vbInformation, "확인")
strQuery = "INSERT INTO MEMBERLIST VALUES "''' & CheckNullData(txtS "''' & CheckNullData(txtS "''' & CheckNullData(txtS "''' & CheckNullData(txtS "''' & CheckNullData(txtS "''' & CheckNullData(txtS	('" & CheckNullData(txtSnum.Text) & "', " & _ name.Text) & "', " & _ sex.Text) & "', " & _ phon.Text) & "', " & _ email.Text) & "', " & _ addr.Text) & "')" 하는 구문을 작성한다.
Else - 같은 회원 코드가 있다면 수정해야 할 것이다. 수정 여 nRtn = MsgBox("데이터를 수정하시겠습니까", vbOKCa If nRtn <> vbOK Then Exit Sub	l부를 확인한다. ancel + vbInformation, "확인")





```
strQuery = "UPDATE MEMBERLIST " & _
          "SET U_SNAME = '" & CheckNullData(txtSname.Text) & "', " & _
            U_SSEX = '" & CheckNullData(txtSsex.Text) & "', " & _
            U_SPHON = " & CheckNullData(txtSphon.Text) & ", " & _
            U_SEMAIL = '" & CheckNullData(txtSemail.Text) & "', " & _
          " U_SADDRESS= '" & CheckNullData(txtSaddr.Text) & "'" & _
          "WHERE U_SNUM = " & CheckNullData(txtSnum.Text) & ""
     - 데이터 베이스 회원 정보 테이블에 회원 정보를 수정하는 구문을 작성한다.
  End If
                                             - 회원 정보 테이블에 레코드를 추가 또는 수정한다.
  conn.Execute strQuery, nCnt
                                             - 데이터 베이스 연결을 종료한다.
  Set conn = Nothing
  strQuery = "SELECT U_SNUM, U_SNAME, U_SSEX, U_SPHON, U_SEMAIL, U_SADDRESS " & _
        "FROM MEMBERLIST ORDER BY U_SNUM"
  - 데이터 베이스 회원 정보 테이블의 회원 정보를 재 검색하는 구문을 작성한다.
  Call SearchData(strQuery)
                                            - 회원 정보를 검색하는 함수를 실행한다.
End Sub
  데이터 베이스 회원 정보 테이블에 회원을 삭제 하기 위해 학번을 입력하고 삭제한다.
Private Sub cmdDelete Click()
  Dim conn As ADODB.Connection
  Dim nCnt, nRtn As Integer
  Dim strQuery As String
 - 메시지 박스로 삭제 여부를 물어보고, OK 을 클릭하면 삭제 구문을 실행하고, Cancel 을 클릭하면 삭제하지 않고
프로시저를 종료한다.
  nRtn = MsgBox("데이터를 삭제하시겠습니까", vbOKCancel + vbInformation, "확인")
  If nRtn <> vbOK Then Exit Sub
  If Not CheckValidate Then Exit Sub
                                            - 필수 입력 데이터가 입력되지 않았는지 확인한다.
  Set conn = New ADODB.Connection
                                             - 데이터 베이스를 연결한다.
  conn.Open connStr
  strQuery = "DELETE FROM MEMBERLIST WHERE U_SNUM = '" & CheckNullData(txtSnum.Text) & "'"
   - 데이터 베이스 회원 정보 테이블에서 학번이 일치하는 회원 정보를 삭제하는 구문을 작성한다.
                                             - 해당 회원 정보 레코드를 삭제한다.
  conn.Execute strQuery, nCnt
                                            - 데이터 베이스 연결을 종료한다.
  Set conn = Nothing
  strQuery = "SELECT U_SNUM, U_SNAME, U_SSEX, U_SPHON, U_SEMAIL, U_SADDRESS " & _
        "FROM MEMBERLIST ORDER BY U_SNUM"
  - 데이터 베이스 회원 정보 테이블의 회원 정보를 재 검색하는 구문을 작성한다.
  Call SearchData(strQuery)
                                            - 회원 정보를 검색하는 함수를 실행한다.
End Sub
  회원 정보 테이블에서 검색하고자 하는 회원 정보를 학번, 이름으로 검색할 수 있는 기능을 추가한다.
Private Sub cmdSearch_Click()
  Dim strQuery As String
  strQuery = "SELECT U_SNUM, U_SNAME, U_SSEX, U_SPHON, U_SEMAIL, U_SADDRESS FROM
MEMBERLIST "
  - 데이터 베이스 회원 정보 테이블의 회원 정보를 재 검색하는 구문을 작성한다.
```

한국폴리텍 IV



If txtFSnum.Text <> "" And txtFSname.Text <> "" 7	Then
strQuery = strQuery & "WHERE U_SNUM Like	e '" & txtFSnum.Text & "%' AND U_SNAME Like '" &
txtFSname.Text & "%' "	- 학번과 이름으로 회원을 검색 할 경우 구문을 추가한다.
ElseIf txtFSnum.Text <> "" Then	
strQuery = strQuery & "WHERE U_SNUM Like '"	'& txtFSnum.Text & "%' "
	- 학번으로 회원을 검색 할 경우 구문을 추가한다.
ElseIf txtFSname.Text <> "" Then	
strQuery = strQuery & "WHERE U_SNAME Like	'" & txtFSname.Text & "%' "
	- 이름으로 회원을 검색 할 경우 구문을 추가한다.
Else	- 조건 없이 검색할 경우 구문을 추가하지 않는다.
End If	
strQuery = strQuery & "ORDER BY U_SNUM"	- 학번 순으로 정렬시킨다.
Call SearchData(strQuery)	- 회원 정보를 검색하는 함수를 실행한다.
Exit Sub	

● 현재 폼을 닫는다.

Private Sub cmdExit_Click() Unload Me End Sub





2.6. 출석 관리 프로그램 만들기

출석 관리 프로그램은 리더기에서 Tag 를 읽으면 시리얼 통신을 통해서 Tag ID(학번) 이 수신된다. 수신된 학번으로 데이터 베이스의 학생 정보 테이블을 검색하여 학번이 일치하는 학생의 학생 정보와 현재 시간을 학생정보 프레임에 보여준다. 학생정보가 화면에 나타나면 출석일자, 출석시간, 학생코드, 학생이름, 출석상태를 리스트에 나타내고 데이터 베이스 출석 테이블에 등록된다.

2.6.1. 메인 폼 만들기

프로그램의 구성으로는 학생정보를 등록, 수정, 삭제 할 수 있는 학생등록 화면과 과목 정보를 등록, 수정, 삭제 할 수 있는 과목등록 폼 그리고 메인 화면이 있다.

메인 화면은 다음과 같이 구성되어 있다.

- 학생 정보를 보여주는 학생 정보 프레임
- · 출석 시간, 지각 시간, 총원, 입실, 지각, 결석 정보를 보여주는 출석 현황 프레임
- 시리얼 통신을 설정하고 연결할 수 있는 COMMUNICATION Port 프레임
- 출석 현황 리스트 및 출석 현황 정보를 파일로 저장하는 출석 현황 프레임으로 구성되어 있다.

■ 출석 관리 프로그램 Ver 1.0.2	
고목 선택 	Communication Port
학생 정보 학 번 : ³⁷⁰⁵⁹⁵⁶⁰⁸⁴ 이 름 : 김홍수 학 과 : 전자공학 연락처 : ⁰¹⁷⁻¹²⁴⁻²³⁴³ E-Mail : ^{jun99@nate.com} 출석 시간 : ¹⁷ 시 ²² 분	출석 현황 리스트 변호 출석일자 출석시간 학변 학생이름 출석상태 1 07-02-01 17:22 3078874900 남하원 혈석 2 07-02-01 17:22 3078874900 남하원 혈석 3 07-02-01 17:22 3078870180 미석호 혈석 4 07-02-01 17:22 3705956084 김홍수 혈석
출석 현황 출석 시간 : 15:30 지각 시간 : 15:50 총 원 : 10 입 실 : 4 지 각 : 0 걸 석 : 10	출석 현황 리스트 저장 리스트 초기화

RFID 13.56MHz 스터디킷을 이용한 출석 관리 프로그램

Main Form

번호	종류	속성	
1	Form	Caption : 출석 관리 프로그램	StartUpPosition : 2 -화면 가운데

한국폴리텍 IV 청주대학



프로그램 메인 화면 로딩 이벤트 프로시저 작성

Private Sub Form_Load()				
On Error GoTo GoErr:				
Call SetEnv	- 폼을 로드하기 전에 기초 설정을 한다.			
Call InitGrid	- 플렉스 그리드 초기 화면을 만든다.			
Call AttenCount	- 총원, 입실자 수, 지각자 수, 결석자 수를 가져온다.			
Call SubjectList	- 과목 리스트를 콤보박스로 보여준다.			
Call SearchData	- 출석 현황을 플렉스 그리드 리스트로 보여준다.			
Exit Sub	- 에러가 없으면 여기서 프로시저 종료			
GoErr:	- 에러 처리			
MsgBox "Form_Load:" & Err.Description	- 에러시 메시지 박스로 함수 이름과 에러 메시지를 보여줌			
End Sub	- 프로시저 종료			
 출석관리 메인폼 로드전 콤보박스에 포트번호를 	입력시키고 DB 파일과 경로명을 변수에 저장시킨다.			
Private Sub SetEnv()				
Dim i As Integer				
-				
cboCommPort.Clear	- 콤보박스의 데이터를 지운다.			
For i = 1 To 16 cboCommPort.AddItem "COM" & Str(i) Next i	- 콤보박스에 COM16 번 까지 데이터를 입력시킨다.			
cboCommPort.ListIndex = 0 strDBFile = App.Path & "WdbWlibrary" connStr = "Provider=Microsoft.Jet.OLEDB.4.0;; connStr = connStr & ":Persist Security Info=Fa	- 콤보박스의 데이터 중 COM1 을 선택한다. - DB 파일 경로와 파일명 저장 Data Source = " & strDBFile & ".MDB" alse" - MDB 파일 OPEN 석정값을 변수에 저장			

End Sub

2.6.2. 출석 현황 프레임 만들기

선택 과목의 출석 시간과 지각처리 시간을 보여주고 총원, 입실자 수, 지각자 수, 결석자 수를 보여주는 출석 현황 프레임은 Label 과 Text Box 로 구성되어 있다.





현재 출석 현황 프레임은 보여주기만 하면 되기 때문에 TexBox 전체 속성 Enabled -> False 로 했다.

번호	종류	속성
1	Frame	Caption : 출석 현황
2	Label	Caption : 출석 시간 : BackStyle : 0 -투명
3	TextBox	Text: 0 Enabled : False
2	TextBox	Text : 비워 둠 Enabled : False
1 1 1		

나머지 Label, TextBox 는 속성이 동일하다.

데이터 베이스에서 선택 과목의 출석 시간과 지각처리 시간을 검색하고, 총원, 입실자 수, 지각자 수, 결석자 수를 검색하여 화면에 보여준다. Public Const ATTENDANCE1 As String = "출석" Public Const ATTENDANCE2 As String = "지각" Public Const ATTENDANCE3 As String = "결석" Public Function AttenCount() On Error GoTo GoErr Dim conn As ADODB.Connection Dim rdSet As ADODB.Recordset Dim strQuery As String, strTemp As String Set conn = New ADODB.Connection Set rdSet = New ADODB.Recordset conn.Open connStr - 데이터 베이스를 연결한다. ----- 총 학생수 가져오기 strQuery = "SELECT COUNT(*) FROM STUDENTINFO" - 학생 정보 테이블에서 총 레코드를 검색한다. If rdSet.State <> adStateClosed Then rdSet.Close - 레코드 셋이 열려 있으면 닫는다. End If rdSet.Open strQuery, connStr, adOpenKeyset - 총 레코드 수를 검색해 레코드 셋을 만든다. - 총 레코드 수를 총원 텍스트 박스로 보여준다. txtAtdsum.Text = rdSet(0)rdSet.Close - 레코드 셋을 닫는다. ----- 입실 학생수 가져오기 strQuery = "SELECT COUNT(*) FROM ATTENDANCELIST" - 출석 테이블에서 총 레코드를 검색한다. If rdSet.State <> adStateClosed Then rdSet.Close - 레코드 셋이 열려 있으면 닫는다. End If rdSet.Open strQuery, connStr, adOpenKeyset - 총 레코드 수를 검색해 레코드 셋을 만든다. - 총 레코드 수를 입실 텍스트 박스로 보여준다. TxtAtdsum1.Text = rdSet(0)rdSet.Close - 레코드 셋을 닫는다. ------ 지각 학생수 가져오기 strQuery = "SELECT COUNT(*) FROM ATTENDANCELIST WHERE A_STATE = "" & ATTENDANCE2 & """ - 출석 테이블에서 출석 상태가 지각인 레코드만 검색한다. If rdSet.State <> adStateClosed Then rdSet.Close - 레코드 셋이 열려 있으면 닫는다. End If rdSet.Open strQuery, connStr, adOpenKeyset - 출석 상태가 지각인 레코드 수를 검색해 레코드 셋을 만든다. _____



		(주)키스컴
N	\mathbf{b}	KISCOM

	TxtAtdsum2.Text = rdSet(0)	- 검색한 레코드 수를 지각 텍스트 박스로 보여준다. - 레코드 세우 다느디
	raset.Close	- 데고드 셋들 된근다.
'_	strQuery = "SELECT COUNT(*) FROM ATTEN	결석 학생수 가져오기 NDANCELIST WHERE A_STATE = '" & ATTENDANCE3 & "'" - 출석 테이블에서 출석 상태가 결석인 레코드만 검색한다.
	If rdSet.State <> adStateClosed Then rdSet.Close End If	- 레코드 셋이 열려 있으면 닫는다.
	rdSet.Open strQuery, connStr, adOpenKeyset strTemp = rdSet(0) rdSet.Close	- 출석 상태가 결석인 레코드 수를 검색해 레코드 셋을 만든다. - 검색한 레코드 수를 임시 변수에 저장한다. - 레코드 셋을 닫는다.
텐	txtAtdnum3.Text = Val(txtAtdsum.Text) - Val - 총원에서 입실자 수를 빼서 미체크 학생수를 스트 반스로 보여준다	(txtAtdnum1.Text) + Val(strTemp) 구하고, 입실자 중에서도 출석 상태가 결석인 학생을 더하여 결석
E:	Set conn = Nothing xit Function	- 데이터 베이스 연결을 종료한다.
2	6.3 과목 선택 프레임 만득기	

• 현재 수강할 과목을 선택하는 과목 선택 프레임은 ComboBox 로 구성되어 있다.

_과목 선택	►	Frame	
	•	ComboBox	

번호	종류	속성
1	Frame	Caption : 과목 선택
2	ComboBox	Text : 비워 둠 Enabled : True

• 데이터 베이스 과목 정보 테이블에서 모든 과목을 검색해 콤보박스에 과목명을 추가하여 보여준다.

Private Function SubjectList() Dim conn As ADODB.Connection Dim rdSet As ADODB.Recordset Dim strQuery As String	
Set conn = New ADODB.Connection Set rdSet = New ADODB.Recordset conn.Open connStr	- 데이터 베이스를 연결한다.
strQuery = "SELECT * FROM COURSEINFO If rdSet.State <> adStateClosed Then rdSet.Close End If	ORDER BY C_NAME ASC" - 과목 정보 테이블에서 모든 레코드를 검색해 이름순 정렬한다. - 레코드 셋이 열려 있으면 닫는다.





rdSet.Open strQuery, connStr, adOpenKeyset cboSubject.Clear	- 과목 정보 테이블의 모든 레코드를 검색해 레코드 셋을 만든다. - 과목 선택 콤보박스의 리스트를 모두 지운다.
If rdSet.RecordCount > 0 Then Do While Not rdSet.EOF cboSubject.AddItem Trim(rdSet("C_NAM rdSet.MoveNext	E")) - 과목명 레코드를 과목 선택 콤보박스에 추가한다. - 다음 레코드로 이동한다.
Loop cboSubject.ListIndex = 0 End If	- 과목 선택 콤보박스의 첫번째 항목을 선택하여 보여준다.
rdSet.Close Set conn = Nothing End Function	- 레코드 셋을 닫는다. - 데이터 베이스 연결을 종료한다.
 출석 관리 프로그램을 실행하면 출석 관리할 과- 	목을 먼저 선택해야한다.
Private Sub cboSubject_Click()	
- 과목선택 콤보박스에서 과목을 선택하면 데이터 Call SubjectSearch(cboSubject.Text)	베이스에서 해당 과목 정보를 검색한다.
End Sub	
 과목을 선택하면 선택 과목 정보를 검색해 출석 	시간과 지각시간을 등록 시킨다.
Private Function SubjectSearch(strSubName As S Dim conn As ADODB.Connection Dim rdSet As ADODB.Recordset Dim strQuery As String	String)
Set conn = New ADODB.Connection Set rdSet = New ADODB.Recordset conn.Open connStr	- 데이터 베이스를 연결한다.
strQuery = "SELECT * FROM COURSEINFO W - 과목 선택 콤보박스에서 선택한 과목명으로 과·	7HERE C_NAME = '" & strSubName & "'" 목 정보 테이블의 해당 과목을 검색한다.
If rdSet.State <> adStateClosed Then rdSet.Close End If	- 레코드 셋이 열려 있으면 닫는다.
rdSet.Open strQuery, connStr, adOpenKeyset	- 과목 정보 테이블에서 해당 과목을 검색해 레코드 셋을 만든다.
<pre>If rdSet.RecordCount > 0 Then strSubjectCode = rdSet("C_CODE") txtStartTime.Text = rdSet("C_STATIME") txtEndTime.Text = rdSet("C_ENDTIME") Else</pre>	- 과목 정보 테이블에 선택한 과목의 정보가 있다면 - 레코드 셋의 과목코드를 변수에 저장한다. - 레코드 셋의 출석 시간을 텍스트 박스로 보여준다. - 레코드 셋의 지각 시간을 텍스트 박스로 보여준다.
MsgBox "등록되지 않은 과목입니다." End If	
rdSet.Close Set conn = Nothing End Function	- 레코드 셋을 닫는다. - 데이터 베이스 연결을 종료한다.





2.6.4. 학생 정보 프레임 만들기

• 학생 정보를 보여주는 학생 정보 프레임은 Label 과 Text Box 로 구성되어 있다.



학생 정보 프레임은 보여주기만 하면 되기 때문에 TexBox 전체 속성 Enabled -> False 로 했다.

번호	종류	속성
1	Frame	Caption : 학생 정보
2	Label	Caption : 학번 : BackStyle : 0 -투명
3	TextBox	Text : 비워 둠 Enabled : False

나머지 Label, TextBox 는 속성이 동일하다.

• 시리얼 통신으로 종료 문자까지 수신하여 Tag ID 를 모두 수신했다면 Tag ID(학번)로 데이터 베이스 학생 리스트 데이블에서 해당 학생의 정보를 검색해야 한다.

Private Function StudentSearch(StudentNum As String) As Boolean

Set conn = New ADODB.Connection Set rdSet = New ADODB.Recordset conn.Open connStr - 데이터 베이스를 연결한다. strQuery = "SELECT * FROM STUDENTINFO WHERE S_NUM = "" & StudentNum & "" - 데이터 베이스 학생 리스트 테이블에서 학번으로 레코드를 검색하는 구문을 작성한다. If rdSet.State <> adStateClosed Then rdSet.Close - 레코드 셋이 열려 있으면 단는다. End If rdSet.Open strQuery, connStr, adOpenKeyset - 학번으로 해당 학생 정보를 검색해 레코드 셋을 작성한다. If rdSet.RecordCount > 0 Then txtStdcode.Text = rdSet("S_NUM") - 레코드 셋의 학번을 텍스트 상자로 보여준다. - 레코드 셋의 이름을 텍스트 상자로 보여준다.	Dim conn As ADODB.Connection Dim rdSet As ADODB.Recordset Dim strQuery As String	
strQuery = "SELECT * FROM STUDENTINFO WHERE S_NUM = '' & StudentNum & '''' - 데이터 베이스 학생 리스트 데이블에서 학번으로 레코드를 검색하는 구문을 작성한다. If rdSet.State <> adStateClosed Then rdSet.Close - 레코드 셋이 열려 있으면 닫는다. End If rdSet.Open strQuery, connStr, adOpenKeyset - 학번으로 해당 학생 정보를 검색해 레코드 셋을 작성한다. If rdSet.RecordCount > 0 Then - 레코드 셋의 학번을 텍스트 상자로 보여준다. txtStdcode.Text = rdSet("S_NUM") - 레코드 셋의 학번을 텍스트 상자로 보여준다. txtStdname.Text = rdSet("S_NAME") - 레코드 셋의 이름을 텍스트 상자로 보여준다.	Set conn = New ADODB.Connection Set rdSet = New ADODB.Recordset conn.Open connStr	- 데이터 베이스를 연결한다.
If rdSet.State <> adStateClosed Then rdSet.Close - 레코드 셋이 열려 있으면 닫는다. End If rdSet.Open strQuery, connStr, adOpenKeyset - 학번으로 해당 학생 정보를 검색해 레코드 셋을 작성한다. If rdSet.RecordCount > 0 Then txtStdcode.Text = rdSet("S_NUM") - 레코드 셋의 학번을 텍스트 상자로 보여준다. txtStdname.Text = rdSet("S_NAME") - 레코드 셋의 이름을 텍스트 상자로 보여준다.	strQuery = "SELECT * FROM STUDENTINFC - 데이터 베이스 학생 리스트 테이블에서 학번의) WHERE S_NUM = ''' & StudentNum & "'" 으로 레코드를 검색하는 구문을 작성한다.
If rdSet.RecordCount > 0 Then txtStdcode.Text = rdSet("S_NUM") - 레코드 셋의 학번을 텍스트 상자로 보여준다. txtStdname.Text = rdSet("S_NAME") - 레코드 셋의 이름을 텍스트 상자로 보여준다.	If rdSet.State <> adStateClosed Then rdSet.Close End If rdSet.Open strQuery, connStr. adOpenKeyset	- 레코드 셋이 열려 있으면 닫는다. - 학번으로 해당 학생 정보를 검색해 레코드 셋을 작성하다.
txtClassName.Text = rdSet("S_CLASS")- 레코드 셋의 학과를 텍스트 상자로 보여준다.txtStdphon.Text = rdSet("S_PHON")- 레코드 셋의 연락처를 텍스트 상자로 보여준다.	If rdSet.RecordCount > 0 Then txtStdcode.Text = rdSet("S_NUM") txtStdname.Text = rdSet("S_NAME") txtClassName.Text = rdSet("S_CLASS") txtStdphon.Text = rdSet("S_PHON")	 레코드 셋의 학번을 텍스트 상자로 보여준다. 레코드 셋의 이름을 텍스트 상자로 보여준다. 레코드 셋의 학과를 텍스트 상자로 보여준다. 레코드 셋의 연락처를 텍스트 상자로 보여준다.





txtStder	nail.Text = rdSet("S_EMAIL")	- 레코드 셋의 E-Mail 을 텍스트 상자로 보여준다.
txtAtten	TimeH.Text = Format(Hour(Time())), "00") - 현재 시간(시)를 텍스트 상자로 보여준다.
txtAtten	TimeM.Text = Format(Minute(Time	e()), "00") - 현재 시간(분)을 텍스트 상자로 보여준다.
Else		- 등록 되지 않은 학번일 경우 False 를 반환하고 함수를 종료한다.
MsgBox	"등록되지 않은 학번입니다."	
rdSet.Cl	ose	- 레코드 셋을 닫는다.
Set com	n = Nothing	- 데이터 베이스 연결을 종료한다.
Student	Search = False	- False 를 반환한다.
Exit Fur	nction	- 함수를 종료한다.
End If		
rdSet.Close	e	- 레코드 셋을 닫는다.
- 이미 출∕ strQuery = - 데이터 ↑	석 체크된 학생인지 검사하기 위해 출식 "SELECT * FROM ATTENDANCEL 베이스 출석 테이블에서 학번으로 레코	덕 테이블을 검사한다. .IST WHERE A_STUCODE = '" & StudentNum & "'" .드를 검색하는 구문을 작성한다.
rdSet.Oper If rdSet.Re rdSet.Cl Set com Student Exit Fur End If	a strQuery, connStr, adOpenKeyset cordCount > 0 Then ose n = Nothing Search = False action	- 학번으로 출석 체크 여부를 검색해 레코드 셋을 작성한다. - 이미 출석 체크된 학생이면 False 를 반환하고 함수를 종료한다. - 레코드 셋을 닫는다. - 데이터 베이스 연결을 종료한다. - False 를 반환한다. - 함수를 종료한다.
Set conn =	Nothing	- 데이터 베이스 연결을 종료한다.
StudentSea	arch = True	- True 를 반환하고 함수를 종료한다.
End Function		

2.6.5. 출석 현황 리스트 및 파일 저장 프레임 만들기

● 출석 현황 정보를 리스트로 보여주고, 파일로 저장할 수 있는 출석 현황 프레임은 CommonDialog , CommandButton, MSFlexGrid 로 구성되어 있다.



63

청주대학



번호	종류	속성
1	Frame	Caption : 비워 둠
2	CommandButton	Caption : 출석 현황 리스트 저장 Style : 1-그래픽
3	CommandButton	Caption : 리스트 초기화 Style : 1-그래픽
4	CommonDialog	
5	MSFlexGrid	

• 출석 현황 리스트를 데이터 베이스의 출석 테이블에 추가하고, 플렉스 그리드에 추가해야 한다.

Private Function AddAttendanceList()

Dim conn As ADODB.Connection

Dim strQuery As String

Dim strDate As String, strTime As String, strStdCode As String, strStdName As String, strAttenState As String

Set conn = New ADODB.Connection conn.Open connStr	- 데이터 베이스를 연결한다.
strDate = Format(Now(), "YY/MM/DD") strTime = txtAttenTimeH.Text & ":" & txtAtte	- 현재 날자를 출석 일자 변수에 저장한다. enTimeM.Text - 출석시간 텍스트 상자의 텍스트를 출석 시간 변수에 저장한다.
strStdCode = txtStdcode.Text strStdName = txtStdname.Text	 학번 텍스트 상자의 텍스트를 학번 변수에 저장한다. 이름 텍스트 상자의 텍스트를 이름 변수에 저장한다.
If txtStartTime.Text > strTime Then strAttenState = ATTENDANCE1	- 체크된 시간이 출석 시간 보다 빠르면 출석 상태가 출석이 된다.
Elself txtEndTime.Text > strTime Then strAttenState = ATTENDANCE2	- 체크된 시간이 지각 시간 보다 빠르면 출석 상태가 지각이 된다.
Else strAttenState = ATTENDANCE3 End If	- 체크된 시간이 지각 시간 보다 늦으면 출석 상태가 결석이 된다.
With MSFlexGrid1	- 출석 현황 리스트에 정보를 추가한다.
.Rows = .Rows + 1 .TextMatrix(.Rows - 1, 0) = .Rows - 1 .TextMatrix(.Rows - 1, 1) = strDate .TextMatrix(.Rows - 1, 2) = strTime .TextMatrix(.Rows - 1, 3) = strStdCode .TextMatrix(.Rows - 1, 4) = strStdName .TextMatrix(.Rows - 1, 5) = strAttenState End With	 리스트 번호를 리스트에 추가한다. 출석 일자을 리스트에 추가한다. 출석 시간을 리스트에 추가한다. 학번을 리스트에 추가한다. 이름을 리스트에 추가한다. 출석 상태를 리스트에 추가한다.
strQuery = "INSERT INTO ATTENDANCELIS VALUES ('" & CheckNullData(strDate) & "', " & "'" & CheckNullDat "''' & CheckNullDat "''' & CheckNullDat "''' & CheckNullDat	ST(A_DATE, A_TIME, A_STUCODE, A_COUCODE, A_STATE)
- 데이터 베이스 술석 테이블에 술석 성보 레코드를	는 수가하는 구분을 삭성한다.
conn.Execute strQuery	- 출석 정보를 출석 테이블에 추가한다.
Set conn = Nothing End Function	- 데이터 베이스 연결을 종료한다.



	(주)키스컴
N	KISCOM

출석 현황 리스트를 데이터 베이스 및 플렉스 그리드에서 모두 삭제한다.

Private Sub cmdAtdlistDelete_Click() Dim conn As ADODB.Connection Dim strQuery As String Set conn = New ADODB.Connection conn.Open connStr - 데이터 베이스를 연결한다.

strQuery = "DELETE FROM ATTENDANCELIST" - 데이터 베이스 출석 테이블의 레코드를 모두 삭제하는 구문을 작성한다.

• 출석 현황 정보를 파일로 저장하기 위해 파일 저장 프로시저를 작성한다.

Private Sub cmdAtdListSave_Click() - 출석 현황 리스트를 파일로 저장한다. Dim strQuery As String, strHead As String

= "SELECT ATTENDANCELIST.A_DATE, ATTENDANCELIST.A_TIME, strQuerv ATTENDANCELIST.A_STUCODE, ATTENDANCELIST.A_COUCODE, ATTENDANCELIST.A_STATE, " & _ "STUDENTINFO.S_NAME, STUDENTINFO.S_CLASS, STUDENTINFO.S_PHON, " & _ "COURSEINFO.C_NAME, COURSEINFO.C_TCHNAME " & _ "FROM (ATTENDANCELIST LEFT JOIN STUDENTINFO ON ATTENDANCELIST.A_STUCODE = STUDENTINFO.S_NUM) " & _ "LEFT JOIN COURSEINFO ON ATTENDANCELIST.A_COUCODE = COURSEINFO.C_CODE" - 데이터 베이스 출석 테이블의 출석 정보 레코드로 학생 정보 테이블과 과목 정보 테이블의 모든 데이터를 검색하는 구문을 작성한다. - 파일 제목을 버튼 Caption 으로 지정한다. strHead = cmdAtdlistSave.Caption - 출석 정보를 파일로 저장한다. Call SaveCsvFile(strQuery, strHead) Call SaveExcelFile(strQuery, strHead) - 출석 정보를 엑셀 파일로 저장한다.

End Sub

• MSFlexGrid 크기를 지정하고 필드명을 입력하여 초기화 한다.

Private Sub InitGrid() With MSFlexGrid1 .Clear

.Rows = 1 .Cols = 6

.FormatString = ">번호|<출석일자|<출석시간|<학생코드|<학생이름|<출석상태" .ColWidth(0) = 500 - 번호 .ColWidth(1) = 1200 - 출석일자

.ColWidth(2) = 1000 - 출석시간 .ColWidth(2) = 2460 - 코드 .ColWidth(4) = 1000 - 학생이름





.ColWidth(5) = 1000 - 출석상태 End With

End Sub





2.6.6. 학생 등록 폼 만들기

학생 등록 화면은 다음과 같이 구성되어 있다.

- 학생 정보를 보여주는 프레임
- 찾고자 하는 학생을 검색할 수 있는 검색 프레임
- 학생 목록을 보여주고 검색 결과를 보여주는 리스트로 구성되어 있다.
- 학생 등록 화면은 메인에서 메뉴를 선택하면 나타난다. 학생 등록 화면에서 필요한 기능은 DB에 등록된 학생들을 리스트에 보여주고 학생을 등록, 수정, 삭제 할 수 있는 기능과 학번, 이름으로 학생을 검색 할 수 있는 기능을 구현해야 한다.



비구	スミ	入山
번오	で 市	<u>4</u> 2
1	Frame	Caption : 비워 둠
1	Frame	Caption : 검색
2	Label	Caption : 학번 : BackStyle : 0 -투명
3	TextBox	Text : 비워 둠
4	CommandButton	Caption : 등록/수정 Style : 1-그래픽
6	MSFlexGrid	
1 1 1		

나머지 Label, TextBox, CommandButton 은 속성이 동일하다.

• 학생 등록 폼이 실행되면 학생 목록전체를 검색하기 위해 데이터 베이스 위치 및 파일명을 변수에 저장한다.



Rev.1.0



68

한국폴리텍 IV

```
Private Sub Form_Load()
  Dim strQuery As String
  strDBFile = App.Path & "₩db₩library"
                                             - DB 파일 경로와 파일명을 변수에 저장한다.
  connStr = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source = " & strDBFile & ".MDB"
  connStr = connStr & ";Persist Security Info=False" - DB 파일 OPEN 설정 값을 변수에 저장한다.
  Call InitGrid
                                              - 그리드를 초기화 한다.
   strQuery = "SELECT S_NUM, S_NAME, S_CLASS, S_PHON, S_EMAIL " & _
                    "FROM STUDENTINFO ORDER BY S_NUM"
   - 데이터 베이스 학생정보 테이블의 학생 정보를 검색하는 구문을 작성한다.
                                              - 학생 정보를 검색하는 함수를 실행한다.
  Call SearchData(strQuery)
End Sub
   학생 목록을 리스트에 보여주기 위해 먼저 리스트의 필드 수와 필드 명을 입력한다.
Private Sub InitGrid()
  With MSFlexGrid1
                                              - 플렉스 그리드를 모두 지운다.
    .Clear
    .Rows = 1
    .Cols = 6
                                              - 필드 이름과 크기를 정한다.
    .FormatString = ">번호|^학번|<이름|<학과|<연락처|<E-Mail"
    .ColWidth(0) = 450
                         '번호
                         '학번
    .ColWidth(1) = 1800
                         '이름
    .ColWidth(2) = 1000
                         '학과
    .ColWidth(3) = 1500
                         '연락처
    .ColWidth(4) = 1300
    .ColWidth(5) = 1300
                         'E-Mail
  End With
End Sub
   데이터 베이스에서 학생정보 테이블 전체를 읽어와 플렉스 그리드로 보여준다.
Private Function SearchData(strQuery As String)
  Dim conn As ADODB.Connection
  Dim rdSet As ADODB.Recordset
  Dim nCnt, nRtn As Integer
                                              - 플렉스 그리드를 초기화 한다.
  Call InitGrid
  Set conn = New ADODB.Connection
  Set rdSet = New ADODB.Recordset
  conn.Open connStr
                                              - 데이터 베이스를 연결한다.
  rdSet.Open strQuery, connStr, adOpenKeyset
                                              - 새로운 레코드 셋을 작성한다.
  If rdSet.RecordCount > 0 Then
                                              - 레코드 셋에 레코드가 있다면
    While Not rdSet.EOF
                                              - 레코드 수 만큼 데이터를 플렉스 그리드에 추가한다.
      With MSFlexGrid1
         .Rows = .Rows + 1
         .TextMatrix(.Rows - 1, 0) = .Rows - 1
                                                     - 리스트 번호를 리스트에 추가한다.
         .TextMatrix(.Rows - 1, 1) = rdSet("S_NUM")
                                                      - 학번을 리스트에 추가한다.
         .TextMatrix(.Rows - 1, 2) = rdSet("S_NAME")
                                                      - 이름을 리스트에 추가한다.
```



- 학과를 리스트에 추가한다. .TextMatrix(.Rows - 1, 3) = rdSet("S_CLASS") - 연락처를 리스트에 추가한다. .TextMatrix(.Rows - 1, 4) = rdSet("S_PHON") .TextMatrix(.Rows - 1, 5) = rdSet("S_EMAIL") - E-Mail 을 리스트에 추가한다. End With - 다음 레코드로 이동한다. rdSet.MoveNext Wend Else MsgBox "등록된 데이터가 없습니다" - 검색된 레코드가 없다면 메시지 박스를 띄운다. End If rdSet.Close - 레코드 셋을 닫는다. Set conn = Nothing - 데이터 베이스 연결을 종료한다. End Function 데이터 베이스 학생정보 테이블에 학생을 등록, 수정 하기 위해 학생 정보를 입력하고 등록 또는 수정한다. Private Sub cmdReg_Click() Dim conn As ADODB.Connection Dim rdSet As ADODB.Recordset Dim nCnt, nRtn As Integer Dim strQuery As String If Not CheckValidate Then Exit Sub - 필수 입력 데이터가 입력되지 않았는지 확인한다. Set conn = New ADODB.Connection Set rdSet = New ADODB.Recordset conn.Open connStr - 데이터 베이스를 연결한다. strQuery = "SELECT * FROM STUDENTINFO WHERE S_NUM = '" & Trim(txtStdcode.Text) & "'" - 데이터 베이스 학생정보 테이블에 같은 학생코드를 가진 레코드가 있는지 검색하는 구문을 작성한다. If rdSet.State <> adStateClosed Then rdSet.Close - 레코드 셋이 열려 있으면 닫는다. End If - 새로운 레코드 셋을 작성한다. rdSet.Open strQuery, connStr, adOpenKeyset If rdSet.BOF Or rdSet.EOF Then - 같은 학번이 없다면 - 메시지 박스로 등록 여부를 물어보고, OK을 클릭하면 등록 구문을 실행하고, Cancel을 클릭하면 등록하지 않고 프로시저를 종료한다. nRtn = MsgBox("데이터를 등록하시겠습니까", vbOKCancel + vbInformation, "확인") If nRtn <> vbOK Then Exit Sub End If strQuery = "INSERT INTO STUDENTINFO VALUES ('" & CheckNullData(txtStdcode.Text) & "', " & _ "'" & CheckNullData(txtStdname.Text) & "', " & _ "'" & CheckNullData(txtClassName.Text) & "', " & _ "'" & CheckNullData(txtStdphon.Text) & "', " & _ "'" & CheckNullData(txtStdemail.Text) & "')" - 데이터 베이스 학생정보 테이블에 학생정보를 추가하는 구문을 작성한다. Else - 같은 학생 코드가 있다면 수정해야 할 것이다. 수정 여부를 확인한다. nRtn = MsgBox("데이터를 수정하시겠습니까", vbOKCancel + vbInformation, "확인") If nRtn <> vbOK Then Exit Sub End If strQuery = "UPDATE STUDENTINFO " & _





조립 키트 매뉴얼

"SET S_NAME = " & CheckNullData(txtStdname.Text) & "', " & _ " S_CLASS = '" & CheckNullData(txtClassName.Text) & "', " & _ " S_PHON = '" & CheckNullData(txtStdphon.Text) & "', " & _ " S_EMAIL = "" & CheckNullData(txtStdemail.Text) & """ & _ "WHERE S_NUM = '" & CheckNullData(txtStdcode.Text) & "'" - 데이터 베이스 학생정보 테이블에 학생정보를 수정하는 구문을 작성한다. End If conn.Execute strQuery, nCnt - 학생정보 레코드를 테이블에 추가 또는 수정한다. Set conn = Nothing - 데이터 베이스 연결을 종료한다. strQuery = "SELECT S_NUM, S_NAME, S_CLASS, S_PHON, S_EMAIL " & _ "FROM STUDENTINFO ORDER BY S_NUM" - 데이터 베이스 학생정보 테이블의 학생 정보를 재 검색하는 구문을 작성한다. Call SearchData(strQuery) - 학생 정보를 검색하는 함수를 실행한다. End Sub 데이터 베이스 학생정보 테이블에 학생을 삭제 하기 위해 학생 코드를 입력하고 삭제한다. Private Sub cmdDelete_Click() Dim conn As ADODB.Connection Dim nCnt, nRtn As Integer Dim strQuery As String - 메시지 박스로 삭제 여부를 물어보고, OK 을 클릭하면 삭제 구문을 실행하고, Cancel 을 클릭하면 삭제하지 않고 프로시저를 종료한다. nRtn = MsgBox("데이터를 삭제하시겠습니까", vbOKCancel + vbInformation, "확인") If nRtn <> vbOK Then Exit Sub End If Set conn = New ADODB.Connection conn.Open connStr strQuery = "DELETE FROM STUDENTINFO WHERE S_NUM = '" & CheckNullData(txtStdcode.Text) & "'" - 데이터 베이스 학생정보 테이블에서 학번이 일치하는 학생 정보를 삭제하는 구문을 작성한다. - 해당 학생정보 레코드를 삭제한다. conn.Execute strQuery, nCnt Set conn = Nothing - 데이터 베이스 연결을 종료한다. strQuery = "SELECT C_CODE, C_NAME, C_TCHNAME, C_STATIME, C_ENDTIME " & _ "FROM COURSEINFO ORDER BY C_CODE" - 데이터 베이스 학생정보 테이블의 학생 정보를 재 검색하는 구문을 작성한다. - 학생 정보를 검색하는 함수를 실행한다. Call SearchData(strQuery) End Sub 학생정보 테이블에서 검색하고자 하는 학생정보를 학번 이름으로 검색할 수 있는 기능을 추가한다. Private Sub cmdSearch_Click() Dim strQuery As String Dim intNum As Integer intNum = 0한국폴리텍 IV



```
strQuery = "SELECT S_NUM, S_NAME, S_CLASS, S_PHON, S_EMAIL FROM STUDENTINFO "
  - 데이터 베이스 학생정보 테이블의 학생 정보를 재 검색하는 구문을 작성한다.
  If txtFStdcode.Text <> "" And txtFStdname.Text <> "" Then
       strQuery = strQuery & "WHERE S_NUM Like '" & txtFStdcode.Text & "%' AND S_NAME Like '" &
  txtFStdname.Text & "%' "
                                       - 학번과 이름으로 학생정보를 검색 할 경우 구문을 추가한다.
    ElseIf txtFStdcode.Text <> "" Then
                                       - 학번으로 학생정보를 검색 할 경우 구문을 추가한다.
       strQuery = strQuery & "WHERE S_NUM Like '" & txtFStdcode.Text & "%' "
    ElseIf txtFStdname.Text <> "" Then
                                  - 이름으로 학생정보를 검색 할 경우 구문을 추가한다.
       strQuery = strQuery & "WHERE S_NAME Like '" & txtFStdname.Text & "%' '
    Else
  End If
      strQuery = strQuery & "ORDER BY S_NUM" - 학번 순으로 정렬시킨다.
                                       - 학생 정보를 검색하는 함수를 실행한다.
  Call SearchData(strQuery)
End Sub
   학번, 이름 입력 텍스트 박스에서 엔터키를 눌러 검색할 수 있는 기능을 추가한다.
Private Sub txtFStdcode_KeyDown(KeyCode As Integer, Shift As Integer)
  If KeyCode = 13 Then
    Call cmdSearch_Click
  End If
End Sub
   데이터의 길이로 필수 텍스트 상자에 입력 여부를 알아보고 입력되지 않은 곳에 포커스를 주고 종료한다.
Private Function CheckValidate() As Boolean
 If Len(txtStdcode.Text) = 0 Then
    txtStdcode.SetFocus
```

```
CheckValidate = False
Exit Function
End If
```

Exit Function

MsgBox "학번를 입력하세요" CheckValidate = False

If Len(txtStdname.Text) = 0 Then txtStdname.SetFocus MsgBox "이름을 입력하세요"

End If

End Function





2.6.7. 과목 등록 폼 만들기

과목 등록 화면은 다음과 같이 구성되어 있다.

- 과목 정보를 보여주는 프레임
- 찾고자 하는 과목을 검색할 수 있는 검색 프레임
- 과목 목록을 보여주고 검색 결과를 보여주는 리스트로 구성되어 있다.
- 과목 등록 화면은 메인에서 메뉴를 선택하면 나타난다. 과목 등록 화면에서 필요한 기능은 DB에 등록된 과목들을 리스트에 보여주고 과목을 등록, 수정, 삭제 할 수 있는 기능과 과목코드, 과목명으로 과목을 검색 할 수 있는 기능이 필요하다.



번호	종류	속성
1	Frame	Caption : 비워 둠
1	Frame	Caption : 검색
2	Label	Caption : 과목코드 : BackStyle : 0 -투명
3	TextBox	Text : 비워 둠
4	CommandButton	Caption : 등록/수정 Style : 1-그래픽
6	MSFlexGrid	

나머지 Label, TextBox, CommandButton 은 속성이 동일하다.

• 과목 등록 폼이 실행되면 과목 목록전체를 검색하기 위한 데이터 베이스 위치 및 파일명을 변수에 저장한다.




73

```
Private Sub Form_Load()
  Dim strQuery As String
 strDBFile = App.Path & "₩db₩library"
                                            - DB 파일 경로와 파일명을 변수에 저장한다.
 connStr = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source = " & strDBFile & ".MDB"
 connStr = connStr & ";Persist Security Info=False" - DB 파일 OPEN 설정 값을 변수에 저장한다.
 Call InitGrid
                                              - 그리드를 초기화 한다.
  strQuery = "SELECT C_CODE, C_NAME, C_TCHNAME, C_STATIME, C_ENDTIME " & _
        "FROM COURSEINFO ORDER BY C_CODE"
  - 데이터 베이스 과목정보 테이블의 필요한 과목 정보를 검색하는 구문을 작성한다.
                                              - 과목 정보를 검색한다.
  Call SearchData(strQuery)
End Sub
  과목 목록을 리스트에 보여주기 위해 먼저 리스트의 필드 수와 필드 명을 입력한다.
Private Sub InitGrid()
  With MSFlexGrid1
                                              - 플렉스 그리드를 모두 지운다.
    .Clear
    .Rows = 1
    .Cols = 6
                                              - 필드 이름과 크기를 정한다.
    .FormatString = ">번호|<과목코드|<과목명|<담당교수|<출석시간|<지각시간"
    .ColWidth(0) = 450
                        '번호
                        '과목코드
    .ColWidth(1) = 1000
    .ColWidth(2) = 2000
                        '과목명
                        '담당교수
    .ColWidth(3) = 1500
                        '출석시간
    .ColWidth(4) = 1200
    .ColWidth(5) = 1200
                        '지각시간
  End With
End Sub
   데이터 베이스에서 과목정보 테이블 전체를 읽어와 플렉스 그리드로 보여준다.
Private Function SearchData(strQuery As String)
  Dim conn As ADODB.Connection
  Dim rdSet As ADODB.Recordset
  Dim nCnt, nRtn As Integer
                                              - 플렉스 그리드를 초기화 한다.
  Call InitGrid
  Set conn = New ADODB.Connection
  Set rdSet = New ADODB.Recordset
  conn.Open connStr
                                              - 데이터 베이스를 연결한다.
  rdSet.Open strQuery, connStr, adOpenKeyset
                                             - 새로운 레코드 셋을 작성한다.
  If rdSet.RecordCount > 0 Then
                                             - 레코드 셋에 레코드가 있다면
    While Not rdSet.EOF
                                              - 레코드 수 만큼 데이터를 플렉스 그리드에 추가한다.
      With MSFlexGrid1
        .Rows = .Rows + 1
         .TextMatrix(.Rows - 1, 0) = .Rows - 1
                                                     - 리스트 번호를 리스트에 추가한다.
         .TextMatrix(.Rows - 1, 1) = rdSet("C_CODE")
                                                     - 과목 코드를 리스트에 추가한다.
        .TextMatrix(.Rows - 1, 2) = rdSet("C_NAME")
                                                     - 과목명을 리스트에 추가한다.
```





.TextMatrix(.Rows - 1, 3) = rdSet("C_TCHNA .TextMatrix(.Rows - 1, 4) = rdSet("C_STATI	AME") - 담당교수를 리스트에 추가한다. ME") - 출석 시간을 리스트에 추가한다.
.TextMatrix(.Rows - 1, 5) = rdSet("C_ENDTI	ME") - 지각 시간을 리스트에 추가한다.
rdSet.MoveNext	- 다음 레코드로 이동한다.
Wend	
Else MsgBox "등록된 데이터가 없습니다"	- 검색된 레코드가 없다면 메시지 박스를 띄운다.
End If rdSat Class	- 레ə드 세으 다느다
Set conn = Nothing	- 데이터 베이스 연결을 종료한다.
End Function	
● 데이터 베이스 과목정보 테이블에 과목을 등록, 수정 하기 위해 과목 정보를 입력하고 등록 또는 수정한다.	
Private Sub cmdReg_Click()	
Dim conn As ADODB.Connection	
Dim rCnt. nRtn As Integer	
Dim strQuery As String	
If Not CheckValidate Then Exit Sub	- 필수 입력 데이터가 입력되지 않았는지 확인한다.
Set conn = New ADODB.Connection	
Set rdSet = New ADODB.Recordset	- 레이터 - 레이크 - 슈퍼크리
conn.Open connStr	- 데이터 베이스들 연결한다.
strQuery = "SELECT * FROM COURSEINFO WHERE C_CODE = '" & Trim(txtSubjectcode.Text) & "'" - 데이터 베이스 과목정보 테이블에 같은 과목코드를 가진 레코드가 있는지 검색하는 구문을 작성한다.	
If rdSet.State <> adStateClosed Then	
rdSet.Close	- 레코드 셋이 열려 있으면 닫는다.
rdSet.Open strQuery, connStr, adOpenKeyset	- 새로운 레코드 셋을 작성한다.
If rdSet.BOF Or rdSet.EOF Then	- 같은 과목 코드가 없다면
- 메시지 박스로 등록 여부를 물어보고, OK 을 클릭하면 프리시저를 조리하다	등록 구문을 실행하고, Cancel 을 클릭하면 등록하지 않고
nRtn = MsgBox("데이터를 등록하시겠습니까", vbOK(If nRtn <> vbOK Then	Cancel + vbInformation, "확인")
Exit Sub	
End If	C (III. 9. Charalle N. IIID at a (trat Cubic at a side Trans) 9. III. II. 9
strQuery = "INSERT INTO COURSEINFO VALUES (" & CheckNullData(txtSubjectcode.lext) & ", " & _ """ & CheckNullData(txtSbiname.Text) & "', " &	
"'" & CheckNullData(txtTchName.Text) & "', " & _	
"'" & CheckNullDat	ta(Format(txtStaTimeH.Text, "00") & ":" &
Format(txtStaTimeM.Tex "''' & CheckNullDat	ct, "00")) & "', " & _ ra(Format(txtEndTimeH Text "00") & ":" &
Format(txtEndTimeM.Te:	xt, "00")) & ")"
- 데이터 베이스 과목정보 테이블에 과목정보를 추가	하는 구문을 작성한다.
Else	
- 같은 과목 코드가 있다면 수정해야 할 것이다. 수정 여부를 확인한다.	
nRtn = MsgBox("데이터를 수정하시겠습니까", vbOKCancel + vbInformation, "확인")	
It nRtn <> vbOK Then Exit Sub	
End If	

한국폴리텍 IV 청주대학



strQuery = "UPDATE COURSEINFO " & _ "SET C_NAME = '" & CheckNullData(txtSbjname.Text) & "', " & _ C_TCHNAME = " & CheckNullData(txtTchName.Text) & ", " & _ C_STATIME = " & CheckNullData(Format(txtStaTimeH.Text, "00") & ":" & Format(txtStaTimeM.Text, "00")) & "', " & _ C_ENDTIME = " & CheckNullData(Format(txtEndTimeH.Text, "00") & ":" & Format(txtEndTimeM.Text, "00")) & "'" & _ "WHERE C_CODE = '" & CheckNullData(txtSubjectcode.Text) & "'" - 데이터 베이스 과목정보 테이블에 과목정보를 수정하는 구문을 작성한다. End If - 과목정보 레코드를 테이블에 추가 또는 수정한다. conn.Execute strQuery, nCnt - 데이터 베이스 연결을 종료한다. Set conn = Nothing strQuery = "SELECT C_CODE, C_NAME, C_TCHNAME, C_STATIME, C_ENDTIME " & _ "FROM COURSEINFO ORDER BY C CODE" - 데이터 베이스 과목정보 테이블의 과목 정보를 재 검색하는 구문을 작성한다. - 과목 정보를 검색하는 함수를 실행한다. Call SearchData(strQuery) End Sub 데이터 베이스 과목정보 테이블에 과목을 삭제 하기 위해 과목 코드를 입력하고 삭제한다. Private Sub cmdDelete_Click() Dim conn As ADODB.Connection Dim nCnt, nRtn As Integer Dim strQuery As String - 메시지 박스로 삭제 여부를 물어보고, OK을 클릭하면 삭제 구문을 실행하고, Cancel을 클릭하면 삭제하지 않고 프로시저를 종료한다. nRtn = MsgBox("데이터를 삭제하시겠습니까", vbOKCancel + vbInformation, "확인") If nRtn <> vbOK Then Exit Sub End If Set conn = New ADODB.Connection conn.Open connStr strQuery = "DELETE FROM COURSEINFO WHERE C_CODE = '" & CheckNullData(txtSubjectcode.Text) & "" - 데이터 베이스 과목정보 테이블에서 과목코드가 일치하는 과목 정보를 삭제하는 구문을 작성한다. - 해당 과목정보 레코드를 삭제한다. conn.Execute strQuery, nCnt - 데이터 베이스 연결을 종료한다. Set conn = Nothing strQuery = "SELECT C_CODE, C_NAME, C_TCHNAME, C_STATIME, C_ENDTIME " & _ "FROM COURSEINFO ORDER BY C_CODE" - 데이터 베이스 과목정보 테이블의 과목 정보를 재 검색하는 구문을 작성한다. Call SearchData(strQuery) - 과목 정보를 검색하는 함수를 실행한다. End Sub 과목정보 테이블에서 검색하고자 하는 과목정보를 과목코드, 과목명으로 검색할 수 있는 기능을 추가한다. Private Sub cmdSearch_Click() 한국폴리텍 IV



조립 키트 매뉴얼

Dim strQuery As String Dim intNum As Integer intNum = 0

strQuery = "SELECT C_CODE, C_NAME, C_TCHNAME, C_STATIME, C_ENDTIME FROM COURSEINFO " - 데이터 베이스 과목정보 테이블의 필요한 과목 정보를 재 검색하는 구문을 작성한다.

If txtFSbjname.Text <> "" And txtFTchName.Text <> "" Then

End Sub

• 과목코드, 과목명 입력 텍스트 박스에서 엔터키를 눌러 검색할 수 있는 기능을 추가한다.

Private Sub txtSubjectcode _KeyDown(KeyCode As Integer, Shift As Integer) If KeyCode = 13 Then Call cmdSearch_Click End If End Sub

• 데이터의 길이로 필수 텍스트 상자에 입력 여부를 알아보고 입력되지 않은 곳에 포커스를 주고 종료한다.

Private Function CheckValidate() As Boolean

If Len(txtSubjectcode.Text) = 0 Then txtSubjectcode.SetFocus MsgBox "과목코드를 입력하세요" CheckValidate = False Exit Function End If

If Len(txtSbjname.Text) = 0 Then txtSbjname.SetFocus MsgBox "과목명을 입력하세요" CheckValidate = False Exit Function End If

End Function



76