

스마트 웹 & 콘텐츠 개발(JAVA, 안드로이드앱) 과정

직원 복지 프로그램

포트폴리오 # 02

2019.06.03(월)

김문선

목차

1. 프로그램 소개

2. 요구사항 분석

3. 프로그램 구조

4. 순서도

5. UML

6. 소스코드

7. 실행화면

8. 후기

1. 프로그램 소개

귀사가 개발한 [직원복지프로그램]은 회사에서 필요한 직원들을 관리 및 복지를 케어할 수 있는 프로그램입니다. 근로계약서 내용을 확인, 수정할 수 있으며 직원들을 등록하거나 조회, 삭제할 수 있으며, 급여, 환율 계산 및 쉬어가기 기능을 제공합니다.

직원복지프로그램을 통해 회사의 직원들을 손쉽게 관리할 수 있으며 활용력이 뛰어난 프로그램입니다. 출장이 많은 회사에 필요한 기능인 환율계산을 통해 환율을 바로 알 수 있으며 회사에서 회사기준의 시간급여를 입력하면 직원이 일한 시간과 한 달간 몇 일 일했는지 일수를 입력하면 계산해주는 편리성이 있는 프로그램입니다.

2. 요구사항

1) 근로계약서 확인을 보완한 수정기능

직원들이 근로계약서를 확인하려면 인사부를 방문하여 확인해야 하는 불편함이 있었는데, 이를 해소하기 위해 직원복지프로그램에 추가함. 직원이름을 입력하면 회사 내에 있는 데이터베이스에서 이름을 조건으로 검색해 근로계약서를 보여주고 수정이 가능한 기능을 추가함.

2) 직원 조회에서 검색할 수 있는 조건을 추가

직원 조회에서 직원들이 다른 직원을 검색하기 위해서는 그 직원의 사원번호를 따로 알아내야 해서 검색에 대한 불편함이 있음. 위 요구사항을 보완하기 위해서 직원이름, 전화번호, 이메일 만을 알아도 검색을 가능하게 기능을 구현했습니다.

3) 업무시간 외 휴식시간을 즐길 수 있도록 쉬어가기 메뉴 추가

업무시간 외 휴식시간(점심시간, 말은 업무를 다하고 남는 시간 등)을 위해 회사에서 즐길 수 있는 게임을 추가함. 시간 맞추기 게임은 두 명의 직원이 함께 참여할 수 있어 직원 간의 교류(관계)를 향상시킬 수 있도록 할 수 있음.

3. 프로그램 구조

위 순서에서는 [직원복지프로그램]의 구조에 대해 소개하는 부분입니다.

① port_main.java

printCalendar()			
자료형	명칭	기능, 내용	
void	printCalendar	날짜를 출력해주는 기능 ✓ 매개변수 : String msg, Calendar cal ✓ return 값 : 없음	
자료형	변수명	내용	
int	year	몇 년도인지 확인해주는 변수	
int	month	몇 월인지 확인해주는 변수	
int	day	몇 일인지 확인해주는 변수	
int	dayOfWeek	몇 요일인지 확인해주는 변수	
int	hour	몇 시인지 확인해주는 변수	
int	ampm	오전, 오후인지 구별해주는 변수	
int	minute	몇 분인지 확인해주는 변수	
int	second	몇 초인지 확인해주는 변수	
int	millisecond	몇 밀리초인지 확인해주는 변수	
자료형	변수명	내용	초기값
Calendar	now	지금 날짜 출력해주는 변수	없음
int	menuchoice	메인메뉴 선택하는 변수 [입력] ☞ 1: 근로계약서 2:직원 3:급여 4:환율 5:게임	0
port_workagreement	pwa	근로계약서메뉴를 실행해주는 변수	
port_employeeinfomenu	pei	직원관리메뉴를 실행하는 변수	
port_workmoney	pw	급여계산메뉴를 실행하는 변수	
port_exchangemenu	pe	환율계산메뉴를 실행하는 변수	
int	gamechoice	게임종류를 선택하는 변수 [입력]	0
port_game	pg	게임메뉴를 실행하는 변수	
port_CapitalGame	pcg	수도 퀴즈 게임 실행하는 변수	

② port_workagreement.java

자료형	변수명	내용	초기값
String	workday	근로계약서 중 계약기간 변수	null
String	workhour	근로시간 변수	null

String	weekworkday	근무일 입력하는 변수	null
String	workmoneytype	월급? 시급인지 연봉인지확인변수	null
String	money	월급여를 확인하는 변수	
String	name	직원 이름을 입력받는 변수[입력]	
String	doc~doc6	근로계약서내용을 출력하는 변수	
int	doccheck	근로계약서 수정여부를 체크하는 변수	0
changedoc()			
자료형	명칭	기능, 내용	
void	changedoc	근로계약서 내용을 수정하는 변수 ✓ 매개변수 : 없음	
자료형	변수명	내용	
int	choicemun	수정할 문장을 입력하는 변수[입력]	
String	aftercom	수정할 내용을 사용자로부터 입력받는 변수[입력]	

③ port_workmoney.java

salarymoney()		
자료형	명칭	기능, 내용
void	salarymoney	급여계산을 해주는 기능 ✓ 매개변수 : char type, int num
자료형	변수명	내용
Scanner	scan	사용자로부터 입력 받는 변수
double	a	하루 일한 시간을 입력 받는 변수
double	b	한 달 동안 일한 일 수를 입력 받는 변수
port_workmoney2	c	급여계산식을 불러오는 변수

④ port_workmoney1.java

자료형	변수명	내용	초기값
double	minimumwage	회사 기준 최저 시급 입력 변수	8350
void	salarycalc()	급여계산을 해준다는 안내문 출력	
calcmoney()			
자료형	명칭	기능, 내용	
public	calcmoney	급여계산식을 알려주는 기능 ✓ 매개변수 : 없음 ✓ return값: 0.0	

⑤ port_workmoney2.java

자료형	변수명	내용	초기값
Scanner	scan	사용자가 입력을 가능하게 하는 변수	
double	workhour	직원이 일한시간을 입력 받는 변수	1
double	workday	직원이 한 달 일한 일수 입력 변수	1
salarycalc()			
자료형	명칭	기능, 내용	
void	salarycalc	급여계산 시작을 알려주는 메소드 ✓ 매개변수 : 없음 ✓ return 값 : 없음	
calcmoney()			
자료형	명칭	기능, 내용	
double	calcmoney	급여계산 결과를 알려주는 메소드 ✓ 매개변수 : 없음 ✓ return 값 : 계산된 급여 금액	

⑥ port_employeeinfomenu.java

자료형	변수명	내용	초기값
Scanner	input	사용자한테 입력 받는 변수	
BufferedReader	dis	사용자한테 입력 받는 변수	
int	SelectNumber	직원관리 하위메뉴 입력 변수	0
int	emp_Num	사원번호 저장하는 변수	0
String	emp_Name	사원이름 저장하는 변수	" "
int	emp_Phone	사원 전화번호 저장하는 변수	0
String	emp_Email	사원 이메일 저장하는 변수	" "
ArrayList<String>	list1	직원 이름에 대한 ArrayList 선언함	
ArrayList<String>	list2	직원 전화번호 선언함	
ArrayList<String>	list3	직원 이메일주소에 대한 리스트	
int	a,i	for문 순환용 변수	1
int	number	직원 전체 명단 사용 변수	1
int	temp	직원검색조건 선택 변수	0
int	num1	검색시 사용되는 사원번호	
String	s1	검색 시 이름 출력변수	
int	s2	검색 시 전화번호출력변수	
String	s3	검색 시 이메일 출력변수	
String	name1	이름으로 검색 시 사용하는 변수	

int	index	사원번호 외 검색 시, 사원번호 출력	
int	phone1	폰 번호로 검색 시 입력 변수	
String	email1	이메일로 검색 시 입력 변수	
int	number1	삭제할 사원 번호 입력 변수	

⑦ port_exchangemenu.java

exchange()		
자료형	명칭	기능, 내용
public	exchange	환율계산해주는 메소드 ✓ 매개변수 : 없음
자료형	변수명	내용
int	haneulchoice	환율 계산할 나라 선택
int	koreamoney	환율 계산할 돈 입력 받는 변수
double	haneulmoney;	환율계산완료한 변수

⑧ port_exchangemenu2.java

Coin <<interface>>		
자료형	변수명	기능, 내용
double	usa	미국 환율비율 변수
double	japan	일본 환율비율 변수
double	china	중국 환율비율 변수

⑨ port_game.java

game1Lotto()		
자료형	명칭	기능, 내용
void	game1Lotto	로또 게임 실행 ✓ 매개변수 : 없음 ✓ return 값 : 없음
자료형	변수명	내용
HashSet	set	로또 번호를 저장해주는 변수
int	i	for문 순환용 변수
int	num	랜덤 값 로또 번호를 받는 변수
LinkedList	list	로또 번호를 출력해주는 변수
game2Gambling ()		
자료형	명칭	기능, 내용

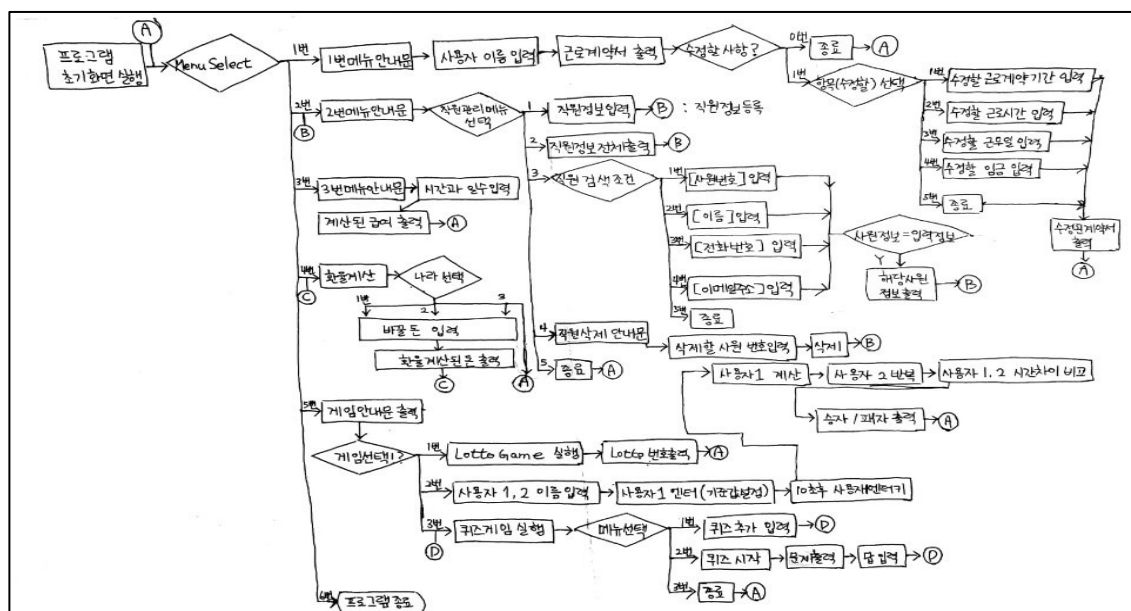
void	game2Gambling	시간맞추기 게임을 실행해주는 메소드 ✓ 매개변수 : 없음 ✓ return 값 : 없음
자료형	변수명	내용
String	name1	게임 사용자 1 이름 입력
String	name2	게임 사용자 2 이름 입력
int	starttime1	사용자1 시작시간
int	endtime1	사용자1 끝 시간
int	result1	사용자1 결과값
int	starttime2	사용자2 시작시간
int	endtime2	사용자2 끝 시간
int	result2	사용자2 결과값

⑩ port_CapitalGame.java

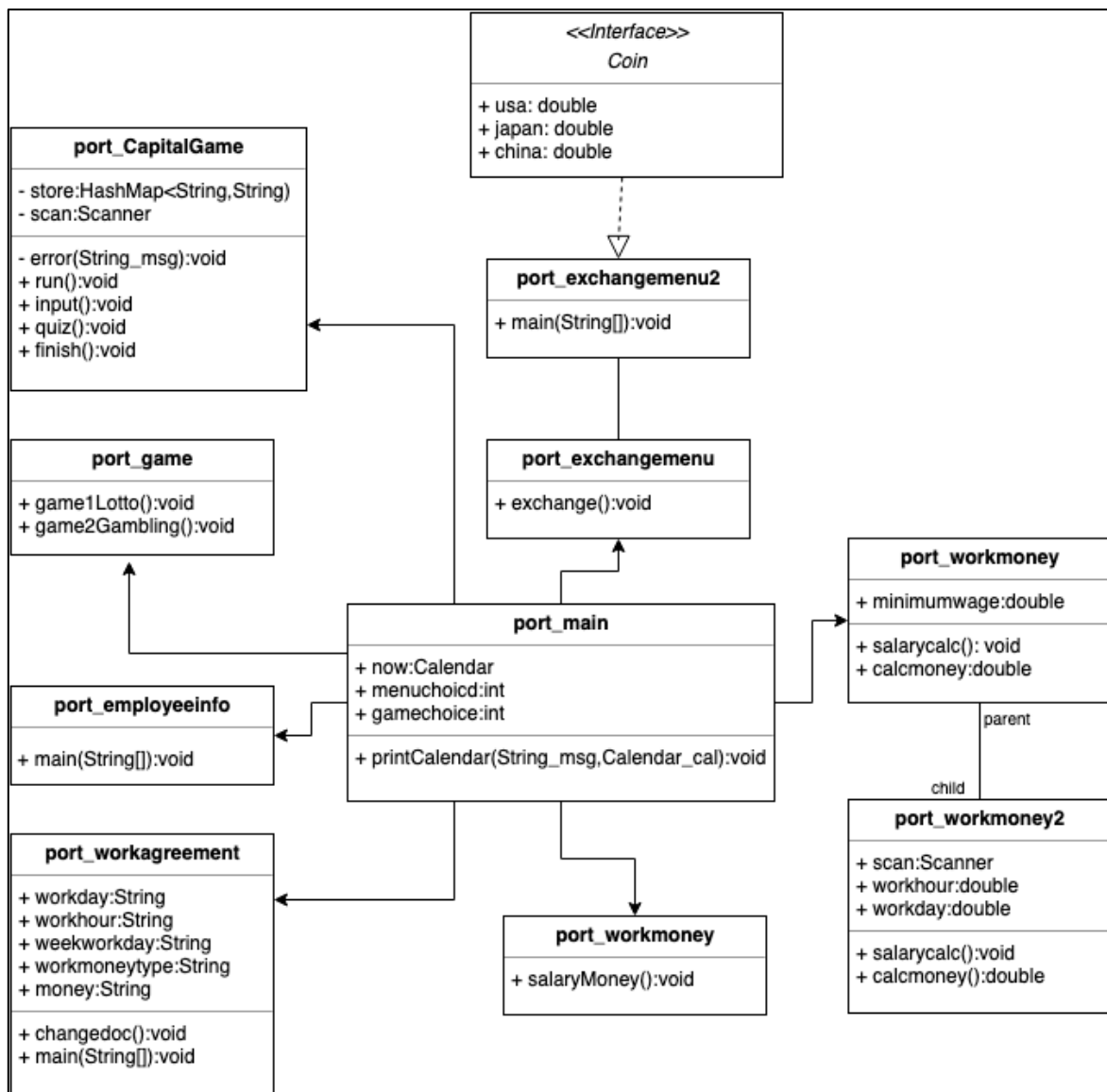
자료형		변수명	내용	초기값
HashMap<String, String>		store	수도 맞추기 게임을 하기위해 저장하는 변수	
port_CapitalGame()				
자료형	함수명	기능, 내용		
public	show	기초적인 문제를 미리 입력해놓음		
error()				
자료형	함수명	기능, 내용		
void	error	메시지를 출력해주는 변수 ✓ 매개변수 : 없음 ✓ return 값 : 없음		
run()				
자료형	명칭	기능, 내용		
void	run	수도맞추기 게임을 실행하는 메소드 ✓ 매개변수 : 없음 ✓ return 값 : 없음		
자료형	변수명	내용		
int	menu	사용자로부터 퀴즈 내용을 추가할 것인지 퀴즈를 실행할지 묻는 변수		
input()				
자료형	명칭	기능, 내용		
void	input	사용자로부터 퀴즈에 내용을 추가해주는 변수 ✓ 매개변수 : 없음		

		✓ return 값 : 없음
자료형	변수명	내용
int	n	입력 받는 맵의 크기 변수
String	country	사용자로부터 입력 받는 나라이름 변수
String	capital	사용자로부터 입력 받는 나라의 수도 변수
quiz()		
자료형	명칭	기능, 내용
void	quiz	입력된 내용을 가지고 퀴즈를 실행해주는 메소드 ✓ 매개변수 : 없음 ✓ return 값 : 없음
자료형	변수명	내용
Set<String>	keys	국가명을 set형태로 변환하는 변수
Object[]	array	set을 배열 형태로 변환해주는 변수
int	index	랜덤한 위치를 결정하는 변수
String	question	번지수를 지정해주는 변수
String	answer	답을 확인하는 변수
String	capitalFromUser	사용자로부터 답을 입력 받는 변수
finish()		
자료형	함수명	기능, 내용
void	finish	게임을 종료하는 메소드 ✓ 매개변수 : 없음 ✓ return 값 : 없음

4. 순서도



5. UML



6. 실행화면

1) 초기화면

회사복지프로그램에오신것을환영합니다

오늘은 2019년 6월2일 일요일 오후8시 58분 55초 입니다.

Plz Select Menu

1근로계약서 확인 및 수정
2직원관리
3급여계산
4환율계산
5쉬어가기
6종료하기

>> 메뉴를 선택해주세요.

2) 근로계약서 확인 화면

>> 메뉴를 선택해주세요. 1
1번 근로계약서 확인및수정메뉴를 선택하셨습니다.
>> 근로자 이름을 입력하세요 : 김문선
=> 김문선님의 근로계약서내용을 출력합니다(회사db에서 불러오므로 3~6초 소요됩니다.)

[[근로 계약서]]
김문선(이하 '갑' 이라 함)과 (주)선주식회사(이하 '을'이라 함)은 다음과 같이 근로계약을 체결한다.
1. 근로계약기간 : 2019.05.25 ~ 2019.10.30
2. 근무장소: 그린빌딩 7층
3. 업무내용: 개발직
4. 근로시간: 09:20 ~ 18:10
5. 근무일/휴일: 주 5일 근무
6. 임금: 월급 2,000,000 원

3) 근로계약서 수정 화면

```
>> 수정사항이 있으신가요(없으면0, 있으면1)??? 1
1번 근로계약서 수정을 선택하셨습니다.
현 시스템 내에 사용가능한 수정기능들을 불러옵니다.
>>
>> 편집 1가지 기능을 제공하고 있으며 다른 것을 편집하고자 하면 인사부로 방문해주시기바랍니다.
*****
          Only 수정 기능만 있습니다!!
*****

수정할 부분을 checking!!
=====
          ①. 근로계약기간
          ②. 근로시간
          ③. 근무일/휴일
          ④. 임금
          ⑤. 종료

# 2.근무장소와 3.업무내용은 수정불가입니다 #
=====
수정할 항목을 선택해주세요(입력) >>1
@ 1번 근로계약기간 항목을 수정합니다.
@ 형식:0000.00.00 ~ 0000.00.00
>> 변경할 기간을 입력해주세요! 2013.03.02~2019.12.31
|
@@ 수정한 내용을 확인하세요!
1. 근로계약기간 :      2013.03.02~2019.12.31
2. 근무장소:          그린빌딩 7층
3. 업무내용:          개발직
4. 근로시간:          09:20 ~ 18:10
5. 근무일/휴일:       주 5일 근무
6. 임금:             월급 2,000,000 원
```

4) 직원관리 중 직원등록 화면

```
>> 메뉴를 선택해주세요. 2
2번 직원관리를 선택하셨습니다!

          [[ 직원 관리 메뉴 ]]
          ----- 보기 -----
          1. 직원 정보 등록
          2. 직원 리스트 출력
          3. 직원 검색
          4. 직원 삭제
          5. 프로그램 종료(exit)
          -----
무엇을 하시겠습니까?
위의 보기에서 해당 번호를 찾아 입력하십시오. >>1

** 직원 정보 등록 **
>> 사원 번호 입력(1~999) 777
>> 직원 이름 입력(한글 2~4자) 김문선
>> 직원 전화번호 입력(ex)01012345678/- 생략):01012345678
>> 직원 이메일 주소를 입력(ex)abc1234@naver.com):jaks111@abcv.com
[직원정보 등록을 완료했습니다!!]
```

5) 직원관리 중 직원 리스트 출력

```
----- 보기 -----
1. 직원 정보 등록
2. 직원 리스트 출력
3. 직원 검색
4. 직원 삭제
5. 프로그램 종료(exit)

-----
무엇을 하시겠습니까?
위의 보기에서 해당 번호를 찾아 입력하십시오. >>2
|
** 직원 리스트 출력 **
(1) 직원번호 : 22 / 이름 : 박시영 / 전화번호 : 1012123434 / 이메일주소 : djdjj@qkdf.com
(2) 직원번호 : 323 / 이름 : 임보라 / 전화번호 : 1022339988 / 이메일주소 : djdj@bkdd.com
(3) 직원번호 : 777 / 이름 : 김문선 / 전화번호 : 1012345678 / 이메일주소 : jaks111@abcv.com
```

6) 직원관리 중 직원 검색

```
----- 보기 -----
1. 직원 정보 등록
2. 직원 리스트 출력
3. 직원 검색
4. 직원 삭제
5. 프로그램 종료(exit)

-----
무엇을 하시겠습니까?
위의 보기에서 해당 번호를 찾아 입력하십시오. >>3

**** 직원 검색 ****

다음 중 원하는 검색 조건을 입력하세요. (1~4중 선택)
1. 사원 번호로 검색
2. 이름으로 검색
3. 전화번호로 검색
4. 이메일 주소로 검색
>> 조건을 입력하세요 : 1
검색할 사원 번호 입력 : 22
< 검색결과 >
사원 번호 : 22 / 사원 이름 : 박시영 / 전화번호 : 1012123434 / 이메일 : djdjj@qkdf.com
```

7) 직원관리 중 직원 삭제

```

----- 보기 -----
1. 직원 정보 등록
2. 직원 리스트 출력
3. 직원 검색
4. 직원 삭제
5. 프로그램 종료(exit)

무엇을 하시겠습니까?
위의 보기에서 해당 번호를 찾아 입력하십시오. >>4

** 직원 삭제 **
삭제하고 싶은 직원의 번호를 입력하세요.
(직원 번호를 모르면 직원 검색 기능을 이용하여 번호를 입력하십시오.)
22
삭제 완료 !

----- 보기 -----
1. 직원 정보 등록
2. 직원 리스트 출력
3. 직원 검색
4. 직원 삭제
5. 프로그램 종료(exit)

무엇을 하시겠습니까?
위의 보기에서 해당 번호를 찾아 입력하십시오. >>2
|
** 직원 리스트 출력 **
(1) 직원번호 : 323 / 이름 : 임보라 / 전화번호 : 1022339988 / 이메일주소 : djdj@bkdd.com
(2) 직원번호 : 777 / 이름 : 김문선 / 전화번호 : 1012345678 / 이메일주소 : jaksl11@abcv.com

```

8) 급여계산

```

Plz Select Menu

1근로계약서 확인 및 수정
2직원관리
3급여계산
4환율계산
5쉬어가기
6종료하기

>> 메뉴를 선택해주세요. 3
5번 급여계산메뉴를 선택하셨습니다.
>> 하루 일한 시간을 입력하세요6
>> 한 달동안 일한 일 수를 입력하세요. 20
급여를 계산합니다. (최저시급은 8,350원)
급여는 1002000.0원 입니다.

```


9) 환율계산

Plz Select Menu

1근로계약서 확인 및 수정
2직원관리
3급여계산
4환율계산
5쉬어가기
6종료하기

>> 메뉴를 선택해주세요. 4

6번 출장가는 직원분들을 위한 서비스 환율계산을 선택하셨습니다!!
환율은 3개의 나라만 제공합니다.

>> 1. 일본 2. 중국 3. 미국

>> 4를 누르면 종료합니다.

>> 나라를 선택해주세요. 2

중국 환율 계산을 선택하셨습니다.

>> 바꿀 돈을 입력하세요. 100

100위안은 17100.0원입니다.

환율은 3개의 나라만 제공합니다.

>> 1. 일본 2. 중국 3. 미국

>> 4를 누르면 종료합니다.

>> 나라를 선택해주세요. 4

환율 메뉴를 종료합니다

10) 쉬어가기 중 로또게임

Plz Select Menu

1근로계약서 확인 및 수정
2직원관리
3급여계산
4환율계산
5쉬어가기
6종료하기

>> 메뉴를 선택해주세요. 5

7번 쉬어가는 메뉴를 선택하셨습니다.
쉬어가는 메뉴에서는 3가지 게임을 선택하실 수 있습니다.

1. 로또 2. 시간맞추기 3. 퀴즈

**4번을 누르면 종료됩니다.

>> 시작할 게임을 선택해주세요. 1

Lotto 게임을 시작합니다!!
이번주 대박을 꿈꾸세요!!
이번주 행운의 번호는 [6, 18, 27, 30, 34, 40] 입니다.

11) 쉬어가기 중 시간맞추기

7번 쉬어가는 메뉴를 선택하셨습니다.
쉬어가는 메뉴에서는 3가지 게임을 선택하실 수 있습니다.

1. 로또 2. 시간맞추기 3. 퀴즈

**4번을 누르면 종료됩니다.

>> 시작할 게임을 선택해주세요. 2

Gambling 게임을 시작합니다!!

이기는 편, 내 편~~!!!

진 사람은 이긴 사람한테 점심 쓰기. ㅋㅋ

>> [Player1] 이름을 입력하세요 김문선

>> [Player2] 이름을 입력하세요 이유경

10초에 가까운 사람이 이기는 게임입니다.

[김문선] 시작 <Enter>키>>

현재 초 시간 = 35

10초 예상 후 <Enter>키>>

현재 초 시간 = 43

[이유경] 시작 <Enter>키>>

현재 초 시간 = 46

10초 예상 후 <Enter>키>>

현재 초 시간 = 54

[김문선] 결과: 8, [이유경] 결과: 8

무승부

♪한치 두치 세치 네치 부꾸빠부꾸뽕~♪♪

♪한치 두치 세치 네치 부꾸부꾸뽕뽕~♪♪

12) 쉬어가기 중 퀴즈

쉬어가는 메뉴에서는 3가지 게임을 선택하실 수 있습니다.

1. 로또 2. 시간맞추기 3. 퀴즈

**4번을 누르면 종료됩니다.

>> 시작할 게임을 선택해주세요. 3

**** 수도 맞추기 게임을 시작합니다. ****

입력:1, 퀴즈:2, 종료:3>> 2

스페인의 수도는? 몽바이

아닙니다!!

영국의 수도는? 런던

정답!!

러시아의 수도는? 러시아

아닙니다!!

그리스의 수도는? 그리스

아닙니다!!

영국의 수도는? 런던

정답!!

중국의 수도는? 북경

아닙니다!!

스페인의 수도는? 그만

7. 소스코드

1) port_main.java

```
import java.util.*;
import java.io.IOException;
import java.sql.Time;
import java.time.*;

public class port_main {
    // 날짜 계산해주는 메소드입니다!!
    public static void printCalendar(String msg, Calendar cal) {
        // 오늘 날짜 계산해주는 부분!!!!
        int year = cal.get(Calendar.YEAR); // 년도
        int month = cal.get(Calendar.MONTH) + 1; // 월
        int day = cal.get(Calendar.DAY_OF_MONTH); // 일
        int dayOfWeek = cal.get(Calendar.DAY_OF_WEEK); //
        int hour = cal.get(Calendar.HOUR); // 시간
        int hourOfDay = cal.get(Calendar.HOUR_OF_DAY);
        int ampm = cal.get(Calendar.AM_PM);
        int minute = cal.get(Calendar.MINUTE);
        int second = cal.get(Calendar.SECOND);
        int millisecond = cal.get(Calendar.MILLISECOND);
        // 년도부터 요일까지 출력!
        System.out.print(msg + year + "년 " + month + "월 " + day + "일 ");
        // 월요일~일요일 출력 안내문.
        switch (dayOfWeek) {
            case Calendar.SUNDAY:
                System.out.print("일요일 ");
                break;
            case Calendar.MONDAY:
                System.out.print("월요일 ");
                break;
            case Calendar.TUESDAY:
                System.out.print("화요일 ");
                break;
            case Calendar.WEDNESDAY:
                System.out.print("수요일 ");
                break;
            case Calendar.THURSDAY:
                System.out.print("목요일 ");
                break;
            case Calendar.FRIDAY:
                System.out.print("금요일 ");
                break;
            case Calendar.SATURDAY:
                System.out.print("토요일 ");
                break;
        }
        // 요일로 출력해주는 case문 끝!
        // 오전인지 오후인지 판별!!
        if (ampm == Calendar.AM) {
            System.out.print("오전");
        } else {
            System.out.print("오후");
        }
        // 현재 시, 분, 초 출력
        System.out.println(hour + "시 " + minute + "분 " + second + "초 입니다.");
    }
    // 오늘 날짜 계산해주는 부분 끝!!!!
}
```

```

// 메인 프로그램
public static void main(String[] args) throws InterruptedException {
    // 프로그램 실행 안내문
    System.out.println(" ");
    System.out.println(" 회사복지프로그램에오신것을환영합니다 ");
    System.out.println(" ");
    // 오늘 날짜 안내 (오늘은 ----년 --월 --일 --요일 --시 --분 --초 입니다.)
    Calendar now = Calendar.getInstance();
    printCalendar("오늘은 ", now);
    System.out.println("*****");

    // \\\\\\\
    Scanner scan = new Scanner(System.in);
    int menuchoice = 0;
    while (menuchoice != 6) {
        System.out.println(" ");
        System.out.println(" Plz Select Menu ");
        System.out.println(" ");
        System.out.println(" 1근로계약서 확인 및 수정 ");
        System.out.println(" 2직원관리 ");
        System.out.println(" 3급여계산 ");
        System.out.println(" 4잔을계산 ");
        System.out.println(" 5쉬어가기 ");
        System.out.println(" 6종료하기 ");
        System.out.println(" ");

        System.out.print(">> 메뉴를 선택해주세요.");
        menuchoice = scan.nextInt();

        switch (menuchoice) {
            case 1: // 근로계약서처럼...근로계약서 //완료!!!!!!
                System.out.println("1번 근로계약서 확인및수정메뉴를 선택하셨습니다.");
                port_workagreement pwa = new port_workagreement();
                pwa.main(args);
                break;

            case 2: // 직원 관리
                System.out.println("2번 직원관리를 선택하셨습니다!");
                port_employeeinfomenu pei = new port_employeeinfomenu();
                pei.main(args);
                break;

            case 3: // 급여계산..... 상속 이음(추상메소드..)와 math에서...
                System.out.println("5번 급여계산메뉴를 선택하셨습니다.");
                port_workmoney pw = new port_workmoney();
                pw.salarymoney();
                break; // 완료!!
        }
    }
}

```

```

case 4: // 환율계산 // interface/ implements 활용
    System.out.println("6번 출장가는 직원분들을 위한 서비스 환율계산을 선택하셨습니다!!");
    port_exchangemenu pe = new port_exchangemenu(); // port_exchangemenu.java에있는거.
    pe.exchange();
    break;
// 완료!!
case 5: // 미니게임(로또게임)과 캠블링...쉬어가는 메뉴..회사퀴즈.....
    System.out.println("7번 쉬어가는 메뉴를 선택하셨습니다.");
    System.out.println("쉬어가는 메뉴에서는 3가지 게임을 선택하실 수 있습니다.");
    System.out.println("-----");
    System.out.println("1.로또      2. 시간맞추기      3. 퀴즈");
    System.out.println("-----");
    System.out.println("**4번을 누르면 종료됩니다.");

    int gamechoice=0;
    System.out.print(">> 시작할 게임을 선택해주세요.");

    gamechoice=scan.nextInt();
    port_game pg = new port_game();
    port_CapitalGame pcg = new port_CapitalGame();

    if(gamechoice==1) { // game 1번 선택
        pg.game1Lotto();
    } else if(gamechoice==2) {
        pg.game2Gambling();
    } else if(gamechoice==3) {

        pcg.run();
    } else if(gamechoice==4) {
        break;
    } else {
        System.out.println("번호를 다시 선택하세요!!");
    }
    break;

case 6: // 프로그램 자체를 종료!!!
    System.out.println("**회사복지프로그램을 종료합니다**");
    break;

default:
    System.out.println("회사복지프로그램에서 제공하는 메뉴는 7번까지입니다...");
    System.out.println("다른 메뉴를 선택해주시기바랍니다...");
    System.out.println("프로그램은 계속 개발하도록 하겠습니다!!!");
    break;
}
}
}
}

```

2) port_workagreement.java

```

import java.io.IOException;
import java.util.*;

public class port_workagreement extends Thread {
    String workday = "2019.05.25 ~ 2019.10.30"; // 근로계약기간
    String workhour = "09:20 ~ 18:10"; // 근무시간
    String weekworkday = "주 5일 근무"; // 5일 근무일 후유일 ex> 주 5일/ 주중
    String workmoneytype = "월급"; // 시급이나? 월급이나? 연봉이나??
    String money = "2,000,000 원"; // 월급금액!!
    //port_workagreement pwa = new port_workagreement();
    //=====근로계약서 수정메뉴

    void changedoc() {
        Scanner scan = new Scanner(System.in);
        System.out.println(">> ");

        System.out.println(">> 편집 1가지 기능을 제공하고 있으며 다른 것을 편집하고자 하면 인사부로 방문해주시기바랍니다.");

        System.out.println("=====");
        System.out.println("only 수정 가능만 있습니다!!");
        System.out.println("=====");
        int choicemun = 0;
        while (choicemun != 5) {
            System.out.println("");
            System.out.println("수정할 부분을 checking!! ");
            System.out.println("=====");
            System.out.println("0. 근로계약기간 ");
            System.out.println("1. 근무시간");
            System.out.println("2. 근무일/주말");
            System.out.println("3. 월급");
            System.out.println("4. 종료");
            System.out.println("");
            System.out.println("# 2.근무장소와 3.업무내용은 수정불가입니다 # ");
            System.out.println("=====");
            System.out.print("수정할 항목을 선택해주세요(입력) >>");

            choicemun = scan.nextInt();
            String aftercom = null; // 사용자가 직접 변경할 내용을 입력받는 변수.

            switch (choicemun) {
                case 1: // 1번을 누르면 근로계약기간을 수정..
                    System.out.println("@ 1번 근로계약기간 항목을 수정합니다.");
                    System.out.println("@ 형식:0000.00.00 ~ 0000.00.00");
                    System.out.print(">> 변경할 기간을 입력해주세요! ");
                    aftercom = scan.next();
                    workday = workday.replace(workday, aftercom);
                    break;
            }
        }
    }
}

```

```

case 2: // 2번을 누르면 근로시간을 수정/
    System.out.println("@ 4번 근로시간을 수정합니다.");
    System.out.println("@ 형식:00:00 ~ 00:00");
    System.out.print(">> 변경할 근로시간을 입력해주세요!");
    aftercom = scan.next();
    workhour = workhour.replace(workhour, aftercom);
    break;

case 3: // 3번을 누르면 5번항목에 대한 근무일,휴일에 대해 수정..!!!
    System.out.println("@ 5. 근무일/휴일 항목을 수정합니다.");
    System.out.print(">> 변경할 근무일/휴일을 입력해주세요!");
    aftercom = scan.next();
    weekworkday = weekworkday.replace(weekworkday, aftercom);
    break;

case 4: // 4번을 누르면 6번 임금항목을 수정!!!
    System.out.println("@ 6. 임금 항목을 수정합니다.");
    System.out.print(">> 변경할 금액(임금)을 입력해주세요!");
    aftercom = scan.next();
    money = money.replace(money, aftercom);
    break;

case 5: // 5번을 누르면 종료가능 실행!!
    System.out.println("근로계약서 수정기능을 종료합니다..");
    break;

default: // 제공하지않는 번호를 눌렀을 경우 반환!!
    System.out.println("수정할 부분을 다시 선택해주세요!!");
    break;
}

// 수정한 내용을 확인하는 차원에서 다시 출력합니다!!!
System.out.println("");
System.out.println("@@ 수정한 내용을 확인하세요!");
String doc, doc0, doc1, doc2, doc3, doc4, doc5, doc6;
doc1 = "1. 근로계약기간 : \t" + workday + "\n";
doc2 = "2. 근무장소: \t 그린빌딩 7층\n";
doc3 = "3. 업무내용: \t 개발직\n";
doc4 = "4. 근로시간: \t" + workhour + "\n";
doc5 = "5. 근무일/휴일: \t" + weekworkday + "\n";
doc6 = "6. 임금: \t" + workmoneytype + " " + money + "\n";
System.out.println(doc1 + doc2 + doc3 + doc4 + doc5 + doc6);
System.out.println("");
}
}

```

```

void main(String[] args) throws InterruptedException {
    // 근로계약서를 불러오기위한 항목으로 thread를 써줍니다
    String workday = "2019.05.25 ~ 2019.10.30"; // 근로계약기간
    String workhour = "09:20 ~ 18:10"; // 근로시간
    String weekworkday = "주 5일 근무"; // 5번 근무일 휴무일 ex) 주 5일/ 주중
    String workmoneytype = "월급"; // 시급이나? 월급이나? 연봉이나??
    String money = "2,000,000 원"; // 월급금액!!

    // 기존의 근로계약서내용을 출력하고....
    Scanner scan = new Scanner(System.in);
    System.out.print(">> 근로자 이름을 입력하세요 : ");
    String name = scan.nextLine();

    // ㅎㅎ스레드 사용..
    System.out.println("==> " + name + "님의 근로계약서내용을 출력합니다(회사db에서 불러오므로 3~6초 소요됩니다.)");
    System.out.println("");
    String doc, doc0, doc1, doc2, doc3, doc4, doc5, doc6;
    doc = "[[ 근로 계약서 ]] \n";
    doc0 = name + "(이하 '갑' 이라 함)과 (주)선주식회사(이하 '을'이라 함)은 다음과 같이 근로계약을 체결한다. \n";
    doc1 = "1. 근로계약기간 : \t" + workday + "\n";
    doc2 = "2. 근무장소: \t 그린빌딩 7층\n";
    doc3 = "3. 업무내용: \t 개발직\n";
    doc4 = "4. 근로시간: \t" + workhour + "\n";
    doc5 = "5. 근무일/휴일: \t" + weekworkday + "\n";
    doc6 = "6. 임금: \t" + workmoneytype + " " + money + "\n";

    // 근로계약서내용을 출력함.
    // 3초 뒤에 thread를 써서 진짜 디비에서 불러오는 것처럼 하기..
    try {
        Thread.sleep(3 * 1000); // 3초 뒤에 근로계약서 내용 불러오는 척하기..
    } catch (InterruptedException e) {
        e.printStackTrace();
    }
    System.out.println(doc + doc0 + doc1 + doc2 + doc3 + doc4 + doc5 + doc6);

    // 기존 근로계약서 작성한것을 보여주고 물어봄....
    // 수정가능이 있는 메뉴 출력..
    int doccheck = 0; // 0이면 완벽하므로 종료!! 1이면 수정선택으로 수정가능 실행
    System.out.println("== 이하 " + name + "님의 근로계약서였습니다 ==");
    System.out.print(">> 수정사항이 있으신가요(없으면0, 있으면1)??? ");

    doccheck = scan.nextInt(); // 사용자로부터 근로계약서에 써져있는 사항이 맞으면 0를 눌러 종료
    // 1을 눌러 수정가능 출력 // 그 외의 것을 누르면 다시반복..!!
}

```


3) port_workmoney.java

```
import java.util.*;

public class port_workmoney {
    void salarymoney() {
        Scanner scan = new Scanner(System.in);
        System.out.print(">> 하루 일한 시간을 입력하세요");
        double a = scan.nextDouble();
        System.out.print(">> 한 달동안 일한 일 수를 입력하세요.");
        double b = scan.nextDouble();

        port_workmoney2 c = new port_workmoney2(a,b);
        c.salarycalc();
        System.out.println("급여는 "+ c.calcmoney()+"원 입니다.");
    }
}
```

4) port_workmoney1.java

```
abstract class port_workmoney1 {
    double minimumwage = 8350; // 최저시급!!!!

    abstract void salarycalc(); // 급여계산

    public double calcmoney() {
        return 0.0;
    }
}
```

5) port_workmoney2.java

```
import java.util.*;

class port_workmoney2 extends port_workmoney1 {
    Scanner scan = new Scanner(System.in);
    double workhour = 1;
    double workday = 1;

    void salarycalc() {
        System.out.println("급여를 계산합니다.(최저시급은 8,350원)");
    }

    public port_workmoney2(double workhour, double workday) { // wh 하루일한 시간
        this.workday = workday;
        this.workhour = workhour;
    }

    public double calcmoney() { // 메서드 오버라이딩
        return minimumwage * workhour * workday;
    }
}
```

6) port_exchangemenu.java

```
import java.util.InputMismatchException;
import java.util.Scanner;

// 환율 계산....
public class port_exchangemenu implements Coin {
    public void exchange() {

        Scanner scan = new Scanner(System.in);
        int haneulchoice = 0;

        int koreamoney; // 환율을 계산할 우리나라돈 입력
        double haneulmoney; // 환율계산완료한 돈

        while (haneulchoice != 4) {
            System.out.println("환율은 3개의 나라만 제공합니다.");
            System.out.println(">> 1. 일본 2. 중국 3. 미국 ");
            System.out.println(">> 4를 누르면 종료합니다.\t");
            System.out.print(">> 나라를 선택해주세요.\t");
            haneulchoice = scan.nextInt(); /// 환율 나라 선택

            switch (haneulchoice) {
                case 1: // 일본 환율 => 100엔이 1000원 :: 1엔이 10원
                    System.out.println("일본 환율 계산을 선택하셨습니다.");
                    System.out.print(">> 바꿀 돈을 입력하세요..\t");
                    // 금액을 입력해야하는데,, 문자입력하면 예외처리발생시키기!!!!
                    try { // try catch문 활용!!
                        koreamoney = scan.nextInt(); // a 3 5
                    } catch (InputMismatchException ie) {
                        System.out.println("정수가 아닙니다. 다시 입력하세요!");
                        scan.next(); // 입력만 받고 저장은 하지 않을!!
                        continue;
                    }

                    haneulmoney = koreamoney * japan;
                    System.out.println(koreamoney + "엔은 " + haneulmoney + "원입니다.");
                    break;

                case 2: // 중국 환율 :: 1위안이 171원
                    System.out.println("중국 환율 계산을 선택하셨습니다.");
                    System.out.print(">> 바꿀 돈을 입력하세요..\t");
                    try { // try catch문 활용!!
                        koreamoney = scan.nextInt(); // a 3 5
                    } catch (InputMismatchException ie) {
                        System.out.println("정수가 아닙니다. 다시 입력하세요!");
                        scan.next(); // 입력만 받고 저장은 하지 않을!!
                        continue;
                    }
                    haneulmoney = koreamoney * china;
                    System.out.println(koreamoney + "위안은 " + haneulmoney + "원입니다.");
                    break;

                case 3: // 미국 환율:: 1달러가 1200원 100달러가 120000
                    System.out.println("미국 환율 계산을 선택하셨습니다.");
                    System.out.print(">> 바꿀 돈을 입력하세요..\t");
                    try { // try catch문 활용!!
                        koreamoney = scan.nextInt(); // a 3 5
                    } catch (InputMismatchException ie) {
                        System.out.println("정수가 아닙니다. 다시 입력하세요!");
                        scan.next(); // 입력만 받고 저장은 하지 않을!!
                        continue;
                    }
                    haneulmoney = koreamoney * usa;
                    System.out.println(koreamoney + "달러는 " + haneulmoney + "원입니다.");
                    break;

                case 4: // 4를 누르면 환율 프로그램 종료 합니다...
                    System.out.println("***환율 메뉴를 종료합니다***");
                    break;

                default:
                    System.out.println("회사에서 환율정보를 제공하는 것은 3개 나라뿐입니다.");
                    System.out.println("일본, 중국, 미국 중 1가지를 선택해주세요.");
                    break;
            }
        }
    }
}
```

7) port_exchangemenu2.java

```
// 환율....
interface Coin { // 환율비를 저장하는 인터페이스.
    double usa = 1200;
    double japan = 10 ;
    double china = 171;
}

public class port_exchangemenu2 {
    public static void main(String[] args) {
        System.out.println("환율은 " + Coin.japan + "원입니다.");
    }
}
```

8) port_game.java

```
public class port_game {
    // 첫번째 게임 로또입니다!!
    void game1Lotto() { // 로또
        System.out.println("Lotto 게임을 시작합니다!!");
        System.out.println("이번주 대박을 꿈꾸세요!!");
        System.out.print("이번주 행운의 번호는 ");
        Set set = new HashSet();

        for (int i = 0; set.size() < 6; i++) {
            int num = (int) (Math.random() * 45) + 1;
            set.add(new Integer(num));
        }

        List list = new LinkedList(set);
        Collections.sort(list);
        System.out.print(list);
        System.out.println(" 입니다.");
    }
}
```

```
// 두 번째 게임링
void game2Gambling() {
    System.out.println();
    System.out.println("Gambling 게임을 시작합니다!!");
    System.out.println("이기는 편, 내 편~!!!");
    System.out.println("진 사람은 이긴 사람한테 점심 쓰기.ㅋㅋ");
    System.out.println("");

    Scanner scan = new Scanner(System.in);
    String name1, name2;
    int starttime1, endtime1, result1, starttime2, endtime2, result2;
    System.out.print(">> [Player1] 이름을 입력하세요");
    name1 = scan.nextLine();
    System.out.print(">> [Player2] 이름을 입력하세요");
    name2 = scan.nextLine();

    System.out.println("10초에 가까운 사람이 이기는 게임입니다.");
    System.out.println("[ " + name1 + " ] 시작 <Enter>키>> ");
    scan.nextLine();
    starttime1 = Calendar.getInstance().get(Calendar.SECOND);
    System.out.println("현재 초 시간 = " + starttime1);
    System.out.println("10초 예상 후 <Enter>키>>");
    scan.nextLine();
    endtime1 = Calendar.getInstance().get(Calendar.SECOND);
    System.out.println("현재 초 시간 = " + endtime1);

    System.out.println("[ " + name2 + " ] 시작 <Enter>키>> ");
    scan.nextLine();
    starttime2 = Calendar.getInstance().get(Calendar.SECOND);
    System.out.println("현재 초 시간 = " + starttime2);
    System.out.println("10초 예상 후 <Enter>키>>");
    scan.nextLine();
    endtime2 = Calendar.getInstance().get(Calendar.SECOND);
    System.out.println("현재 초 시간 = " + endtime2);

    // 결과
    result1 = (endtime1 > starttime1) ? (endtime1 - starttime1) : ((endtime1 + 60) - starttime1);
    result2 = (endtime2 > starttime2) ? (endtime2 - starttime2) : ((endtime2 + 60) - starttime2);
    System.out.println("[ " + name1 + " ] 결과: " + result1 + ", [ " + name2 + " ] 결과: " + result2 + " ");

    if (Math.abs(10 - result1) < Math.abs(10 - result2)) {
        System.out.println("승자는 " + name1 + "!");
    } else if (Math.abs(10 - result1) == Math.abs(10 - result2)) {
        System.out.println("무승부");
    } else {
        System.out.println("승자는 " + name2 + "!");
    }

    System.out.println("♫한치 두지 세지 네지 부구부구뽕~♫\n" + "♫한치 두지 세지 네지 부구부구뽕~♫");
    System.out.println();
}
```


9) port_CapitalGame.java

```
import java.util.HashMap;
import java.util.Iterator;
import java.util.Scanner;
import java.util.Set;

public class port_CapitalGame {
    private HashMap<String, String> store = new HashMap<String, String>();
    private Scanner scanner = new Scanner(System.in);

    public port_CapitalGame() {
        // store에 9 개의 아이템을 입력하여 초기화
        store.put("멕시코", "멕시코시티");
        store.put("스페인", "마드리드");
        store.put("프랑스", "파리");
        store.put("영국", "런던");
        store.put("그리스", "아테네");
        store.put("독일", "베를린");
        store.put("일본", "도쿄");
        store.put("중국", "베이징");
        store.put("러시아", "모스크바");
    }

    private void error(String msg) {
        System.out.println(msg);
    }

    public void run() {
        System.out.println("**** 수도 맞추기 게임을 시작합니다. ****");
        while (true) {
            System.out.print("입력:1, 퀴즈:2, 종료:3>> ");
            int menu = scanner.nextInt();
            switch (menu) {
                case 1:
                    input();
                    break;
                case 2:
                    quiz();
                    break;
                case 3:
                    finish();
                    return;
                default:
                    error("잘못된 입력입니다.");
            }
        }
    }

    private void input() {
        int n = store.size();
        System.out.println("현재 " + n + "개 나라와 수도가 입력되어 있습니다.");
        n++;
        while (true) {
            System.out.print("나라와 수도 입력 " + n + ">> ");
            String country = scanner.next();
            if (country.equals("그만")) {
                break;
            }
            String capital = scanner.next();
            if (store.containsKey(country)) {
                System.out.println(country + "는 이미 있습니다");
                continue;
            }
            store.put(country, capital);
            n++;
        }
    }
}
```

```

private void quiz() {
    Set<String> keys = store.keySet(); // 1. 국가명"키"를-> set형태로
    Object[] array = (keys.toArray()); // 2. set을 배열로 변환

    while (true) {
        // 나라 중에서 하나를 선택한다.
        int index = (int) (Math.random() * array.length); // 랜덤한 위치 결정

        // 문제(나라)와 정답(수도)를 결정한다.
        String question = (String) array[index]; // ex)3번지 : 일본
        String answer = store.get(question); // store.get("일본")

        // 문제를 출력한다.
        System.out.print(question + "의 수도는? ");

        String capitalFromUser = scanner.next(); // 사용자의 입력
        if (capitalFromUser.equals("그만")) {
            break;
        }
        if (capitalFromUser.equals(answer))
            System.out.println("정답!!");
        else
            System.out.println("아닙니다!!");
    }
}

private void finish() {
    System.out.println("게임을 종료합니다.");
}

```

10) port_employeeinfomenu.java

```

// 직원관리메뉴메인입니다!!-----
public class port_employeeinfomenu {
    @SuppressWarnings("deprecation")
    public static void main(String[] args) {

        Scanner input = new Scanner(System.in); // 입력값을 받기 위해 scanner를 사용.
        BufferedReader dis = new BufferedReader(new InputStreamReader(System.in));

        int selectNumber = 0; // 프로그램 시작시 보기에서 선택을 받을.
        // 직원이름과 사원번호, 전화번호, 이메일을 초기화한다.
        int emp_Num = 0;
        String emp_Name = " ";
        int emp_Phone = 0;
        String emp_Email = " ";

        int delete_Num = 0;
        // ArrayList로 직원정보를 등록하기 위해 3개를 선언한다...
        // 배열을 선언해서 ArrayList로 변경해서 사용하려했지만 의문의 빨간줄이 체크된다..
        // 고이고이 접어 하글로 보낸다... 바로 ArrayList를 사용한다.
        // 이름이니까 String
        ArrayList<String> list1 = new ArrayList<String>(); // 직원 이름에 대한 ArrayList 선언함
        // 전화번호니까 Integer를 ...
        ArrayList<Integer> list2 = new ArrayList<Integer>(); // 직원 전화번호에 대한 ArrayList 선언함
        // 이메일이니까 String을 정해주도록..해야한다...!!
        ArrayList<String> list3 = new ArrayList<String>(); // 직원 이메일주소에 대한 ArrayList 선언함
        // 999개의 ArrayList의 방을 초기화 해주는 작업
        // 직원은 딱 1000명만 받기로 한다..
        // 그 이상의 명수는 생략한다.. 1000명도 많다아...
        for (int a = 1; a < 1000; a++) { // 직원정보등록/
            list1.add("null"); // 이름
            list2.add(0); // 전화번호
            list3.add("null"); // 이메일
        }

        System.out.println("");
        System.out.println("    [[ 직원 관리 메뉴 ]]);");

        while (true) { // 무한 루프
            // 직원관리메뉴에 대한 세부메뉴항목을 출력합니다.
            System.out.println("----- 보기 -----");
            System.out.println("    1. 직원 정보 등록");
            System.out.println("    2. 직원 리스트 출력");
            System.out.println("    3. 직원 검색");
            System.out.println("    4. 직원 삭제");
            System.out.println("    5. 프로그램 종료(exit)");
            System.out.println("-----");
            System.out.print("무엇을 하시겠습니까? ");
            System.out.print("위의 보기에서 해당 번호를 찾아 입력하십시오. >>");
            // 입력하십시오....

```



```

        SelectNumber = input.nextInt(); // 직원관리메뉴 안에 있는 메뉴 번호 선택 입력란.
// 1번을 누를경우 출력되는 항목이다. 직원정보를 등록하겠다.
if (SelectNumber == 1) {
    System.out.println("");
    System.out.println("*** 직원 정보 등록 ***");
    // 1.
    while (true) { // while 1번문 사원번호
        System.out.print(">> 사원 번호 입력(1~999)");
        emp_Num = input.nextInt();
        if (emp_Num < 1000 && emp_Num > 0) // 1부터 999사이의 값을 입력하지 않으면 다시 입력받는다.
            break;
    }
    // 2.
    while (true) { // while 2번문 이름
        System.out.print(">> 직원 이름 입력(한글 2~4자)");
        try {
            emp_Name = dis.readLine();
        } catch (IOException e) {
            e.printStackTrace();
        }
        if (emp_Name.length() < 5 && emp_Name.length() > 1)
            // 이름이 2자~4자 입력가능...
            break;
    }
    // 3.
    System.out.print(">> 직원 전화번호 입력(ex)01012345678/- 생략:");
    emp_Phone = input.nextInt();
    // 4.
    System.out.print(">> 직원 이메일 주소를 입력(ex)abc1234@naver.com:");
    emp_Email = input.next();

    list1.set(emp_Num, emp_Name);
    list2.set(emp_Num, emp_Phone);
    list3.set(emp_Num, emp_Email);

    System.out.println("직원정보 등록을 완료했습니다!!");
    System.out.println("");
    continue;
}

// 2번 직원리스트 출력 메뉴이다..
// 2번을 누를 경우, 등록되어있는 모든 직원들이 출력된다.!!
if (SelectNumber == 2) { // 2. 모든 직원 정보 출력
    System.out.println("");
    System.out.println("*** 직원 리스트 출력 ***");

    int number = 1;
    for (int i = 0; i < list1.size(); i++) {
        if (list1.get(i) == "null")
            continue;
        System.out.print("(" + number + ") ");
        number++;
        System.out.print("직원번호 : " + i);
        System.out.print(" / 이름 : " + list1.get(i));
        System.out.print(" / 전화번호 : " + list2.get(i));
        System.out.println(" / 이메일주소 : " + list3.get(i));
    }
    continue;
}
}

```

```
//
if (SelectNumber == 3) { // 3. 직원 검색
    System.out.println("");

    int temp = 0;

    System.out.println("**** 직원 검색 ****");
    System.out.println("");
    System.out.println("다음 중 원하는 검색 조건을 입력하세요. (1~4중 선택)");
    System.out.println("1. 사원 번호로 검색");
    System.out.println("2. 이름으로 검색");
    System.out.println("3. 전화번호로 검색");
    System.out.println("4. 이메일 주소로 검색");
    System.out.print(">> 조건을 입력하세요 : ");
    temp = input.nextInt(); // 어떤 조건을 선택할 것인가 누르세요

    // 1번 사원 번호 조건으로 검색
    if (temp == 1) {
        System.out.print("검색할 사원 번호 입력 : ");
        int num1 = input.nextInt();

        String s1 = list1.get(num1);
        int s2 = list2.get(num1);
        String s3 = list3.get(num1);
        // 검색결과를 출력한다...
        System.out.println("< 검색결과 >");
        System.out.println("사원 번호 : " + num1 + " / 사원 이름 : "
            + s1 + " / 전화번호 : " + s2 + " / 이메일 : " + s3);
    }

    // 2번 이름 조건으로 사원을 검색하게 한다.
    if (temp == 2) {
        System.out.print("검색할 사원 이름 입력: ");

        String name1 = input.next();
        int index = list1.indexOf(name1);
        String s1 = list1.get(index);
        int s2 = list2.get(index);
        String s3 = list3.get(index);
        // 검색결과를 출력한다...
        System.out.println("< 검색결과 >");
        System.out.println("사원 번호 : " + index + " / 사원 이름 : "
            + s1 + " / 전화번호 : " + s2 + " / 이메일 : " + s3);
        System.out.println("");
    }

    // 3번 전화번호로 검색 선택
    if (temp == 3) {
        System.out.print("검색할 사원 전화 번호 입력: ");
        int phone1 = input.nextInt();
        int index = list2.indexOf(phone1);
        String s1 = list1.get(index);
        int s2 = list2.get(index);
        String s3 = list3.get(index);
        // 검색결과를 출력한다...
        System.out.println("< 검색결과 >");
        System.out.println("사원 번호 : " + index + " / 사원 이름 : "
            + s1 + " / 전화번호 : " + s2 + " / 이메일 : " + s3);
        System.out.println("");
    }

    // 4번 이메일 주소로 검색 선택
    if (temp == 4) {
        System.out.print("검색할 사원의 이메일 주소를 입력: ");
        String email1 = input.next();
        int index = list3.indexOf(email1);
        String s1 = list1.get(index);
        int s2 = list2.get(index);
        String s3 = list3.get(index);
        // 검색결과를 출력한다...
        System.out.println("< 검색결과 >");
        System.out.println("사원 번호 : " + index + " / 사원 이름 : "
            + s1 + " / 전화번호 : " + s2 + " / 이메일 : " + s3);
        System.out.println("");
    }

    }
    continue;
} // 3번 직원검색에 대한 항목 if문 종료
```

```

// 4번 직원삭제를 하겠다.. 퇴직하면 있어야되는 메뉴!!
if (SelectNumber == 4) { // 4. 직원 삭제
    System.out.println("");
    System.out.println("*** 직원 삭제 ***");
    System.out.println("삭제하고 싶은 직원의 번호를 입력하세요.");
    System.out.println("(직원 번호를 모르면 직원 검색 기능을 이용하여 번호를 입력하십시오.)");

    int number1 = input.nextInt();

    list1.remove(number1);
    list2.remove(number1);
    list3.remove(number1);
    list1.add(number1, "null"); // remove()를 수행하고 나서 인덱스값이 하나씩 줄어드는 현상을 막기 위함.
    list2.add(number1, 0);
    list3.add(number1, "null");

    // 삭제 완료 !!
    System.out.println("삭제 완료 !!");
    continue;
}
// 5번을 누르면 직원관리메뉴 바2바2!!
if (SelectNumber == 5) {
    break;
}
}
System.out.println("...직원관리프로그램을 종료합니다...");
System.out.println("");
}
}

```

8. 후기

흠... 이번 포트폴리오를 하면서 어떤 주제를 할지 고민을 했습니다.. 상속부터 람다 식까지 배운 걸 다 넣어볼까 생각하면서 만든 프로그램인데,, 결국에는 람다 식은 넣기가 애매하여 안 넣었습니다. 포트폴리오를 작성하면서 '아..코드가 더럽다..'라고 생각했지만, 이미 시간은 6월 3일,, 몇 개만 고치기로 했습니다. 포트폴리오 제작을 위해서 프로그램 제작을 하는 것은 좋은 것이라 생각합니다. 배운걸 되짚어볼 수 있고 실생활에 사용할 수 있어서 좋다고 생각합니다. 수업에서 실습한 코드 재사용한 것도 있지만 재사용하면서 복습을 할 수 있어서 좋았습니다. 더 머리에 쑥쑥 들어가는 기분입니다. 포트폴리오가 이 정도인데, 실제 개발팀으로 가면 머리가 터질 것 같다는 느낌이 들었습니다. 포트폴리오를 하면서 부족한 점을 발견할 수 있는 것 같습니다. 아직까지는 implements 구현하는 것을 헛갈려 하는 것 같습니다. 부족한 부분을 다시 복습해야겠습니다. 재미있습니다. 저만의 프로그램을 한 번 제작해보는 게..! 다음에는 더 창의적인 생각으로 프로그램을 작성해보아야겠습니다... 첫 번째 포트폴리오랑 비슷한 형식으로 제작된 듯 하여.. 3번째 포트폴리오는 생각을 지금부터 해보아야겠습니다.

최종적인 후기는 뿌듯하다, 부족한 부분을 발견할 수 있어서 좋았습니다!!! 좀 더 개발을 잘, 알고리즘을 잘 짤 수 있도록 발전해야겠다는 다짐을 했습니다.