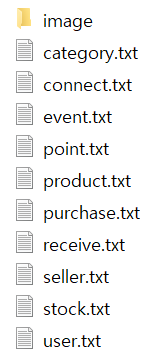
**2021년 전국기능경기대회 과제**

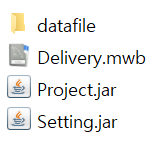
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 직종명 | 정보기술 | 과제명 | Java  Programming I | 과제번호 | 제 2 과제 |
| 경기시간 | 4시간 | 비번호 |  | 심사위원확인 | (인) |

1. 요구사항
2. 준비작업 : 다음의 지시에 따라 경기 진행을 위한 준비 작업을 수행하시오.
   1. 시스템 설치 목록을 확인하시오.

* Eclipse win64
* Jdk-12.0.1\_windows-x64\_bin
* mysql 8.0.16.0
  1. datafile이 다음과 같이 구성되어 있는지 확인하시오.



* 1. 제출 파일 목록



1. 공통 조건
   1. 프로그램의 모든 경고/정보 메시지의 제목과 스타일을 다음과 같이 하시오.

|  |  |
| --- | --- |
| 경고 메세지 | 정보 메세지 |
|  |  |

* 1. 경로, 배송현황, 통계의 확대 창을 제외한 나머지 페이지는 모두 하나의 프레임으로 실행되도록 하시오.

# [문제1] 다음 <조건>을 만족하는 프로그램 ‘Setting.jar’을 작성하시오.

1. Setting.jar을 실행하면 데이터베이스 ‘Delivery'가 생성되게 하시오.   
   (기존에 있었던 데이터베이스일 경우 삭제하고 다시 만드시오.)
2. DB 구성 조건
   1. 모든 문자열은 UTF-8로 설정하시오.
   2. MYSQL 관리자 아이디는 ‘root’ 패스워드는 ‘1234’로 지정하시오.
   3. MYSQL 유저 아이디는 ‘user’ 패스워드는 ‘1234’로 지정하시오.
   4. 데이터베이스의 이름은 ‘Delivery’로 지정하시오.
3. 데이터베이스에 다음과 같은 테이블이 생성되게 하고, 지급자료로 제공한 텍스트 파일들을 테이블에 저장하시오.
   1. point : 맵 상의 모든 지점의 주소 정보

|  |  |  |  |
| --- | --- | --- | --- |
| 필드명 | 데이터 형식 | 속성 | 설명 |
| po\_No | int | PK, AI | 고유번호 |
| po\_X | int |  | X좌표, 주소 앞자리 |
| po\_Y | int |  | Y좌표, 주소 뒷자리 |

* 1. connect : 도로 연결 정보

|  |  |  |  |
| --- | --- | --- | --- |
| 필드명 | 데이터 형식 | 속성 | 설명 |
| c\_Node1 | int | FK | point 고유번호 |
| c\_Node2 | int | FK | point 고유번호 |

* 1. user : 구매자 회원 정보

|  |  |  |  |
| --- | --- | --- | --- |
| 필드명 | 데이터 형식 | 속성 | 설명 |
| u\_No | int | PK, AI | 고유번호 |
| u\_Id | varchar(20) |  | 아이디 |
| u\_Pattern | varchar(10) |  | 비밀번호 (패턴) |
| u\_Name | varchar(20) |  | 이름 |
| u\_Addr | int | FK | point 고유번호 |

* 1. seller : 판매자 회원 정보

|  |  |  |  |
| --- | --- | --- | --- |
| 필드명 | 데이터 형식 | 속성 | 설명 |
| s\_No | int | PK, AI | 고유번호 |
| s\_Id | varchar(20) |  | 아이디 |
| s\_Pattern | varchar(10) |  | 비밀번호 (패턴) |
| s\_Name | varchar(20) |  | 영업점명 |
| s\_Addr | int | FK | point 고유번호 |

* 1. category : 상품 분류 정보

|  |  |  |  |
| --- | --- | --- | --- |
| 필드명 | 데이터 형식 | 속성 | 설명 |
| c\_No | int | PK, AI | 고유번호 |
| c\_Name | varchar(20) |  | 이름 |

* 1. product : 상품 정보

|  |  |  |  |
| --- | --- | --- | --- |
| 필드명 | 데이터 형식 | 속성 | 설명 |
| p\_No | int | PK, AI | 고유번호 |
| c\_No | int | FK | category 고유번호 |
| p\_Name | varchar(50) |  | 이름 |
| p\_Price | int |  | 가격 |

* 1. stock : 상품 재고 정보

|  |  |  |  |
| --- | --- | --- | --- |
| 필드명 | 데이터 형식 | 속성 | 설명 |
| st\_No | int | PK, AI | 고유번호 |
| s\_No | int | FK | seller 고유번호 |
| p\_No | int | FK | product 고유번호 |
| st\_Count | int |  | 개수 |

* 1. event : 행사 정보

|  |  |  |  |
| --- | --- | --- | --- |
| 필드명 | 데이터 형식 | 속성 | 설명 |
| e\_No | int | PK, AI | 고유번호 |
| p\_No | int | FK | product 고유번호 |
| e\_Month | int |  | 월 |

* 1. purchase : 상품 구매 정보

|  |  |  |  |
| --- | --- | --- | --- |
| 필드명 | 데이터 형식 | 속성 | 설명 |
| pu\_No | int | PK, AI | 고유번호 |
| s\_No | int | FK | seller 고유번호 |
| p\_No | int | FK | product 고유번호 |
| u\_No | int | FK | user 고유번호 |
| pu\_Count | int |  | 개수 |
| pu\_Price | int |  | 가격 |
| pu\_Date | date |  | 구매날짜 |

* 1. receive : 배송 정보

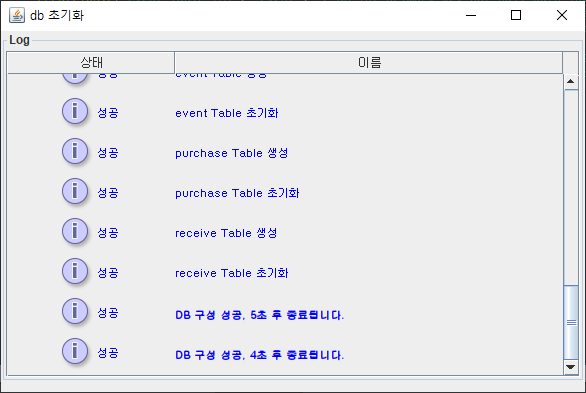
|  |  |  |  |
| --- | --- | --- | --- |
| 필드명 | 데이터 형식 | 속성 | 설명 |
| r\_No | int | PK, AI | 고유번호 |
| pu\_No | int | FK | purchase 고유번호 |
| r\_Time | datetime |  | 배송 시작 날짜/시간 |

1. 회원은 ‘Delivery’ 의 데이터 조회/추가/삭제/업데이트만 가능하도록 하시오. 이외 테이블 생성이나 유저 권한은 제한되게 하시오. 이후 과제의 모든 Connection은 유저로 접속 할 수 있도록 하시오.
2. 실행 도중 처리한 작업에 대해 로그를 표현하시오. ※ Setting.mp4 참조
   1. [표 1]의 순서로 로그가 표시되도록 하시오.

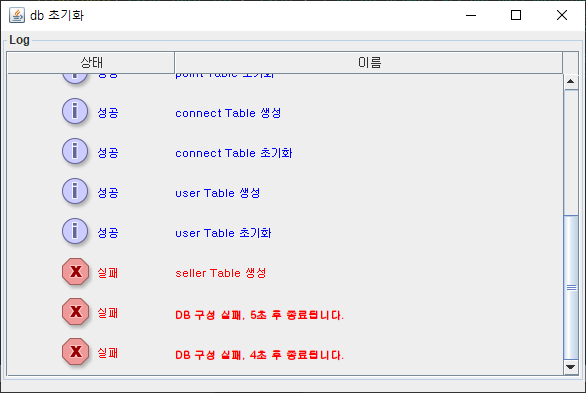
[표 1] 로그

|  |  |
| --- | --- |
| 순서 | 로그 이름 |
| 1 | DB 삭제 |
| 2 | DB 생성 |
| 3 | user 삭제 |
| 4 | user 생성 |
| 5 | user 권한 설정 |
| 6 | point Table 생성 |
| 7 | connect Table 생성 |
| 8 | user Table 생성 |
| 9 | seller Table 생성 |
| 10 | category Table 생성 |
| 11 | product Table 생성 |
| 12 | stock Table 생성 |
| 13 | event Table 생성 |
| 14 | purchase Table 생성 |
| 15 | receive Table 생성 |
| 16 | DB 구성 성공/실패 |
| 17 | ? 초 후 종료됩니다. |

* 1. 모든 초기화 작업이 성공하면 [그림 i]와 같이 “DB 구성 성공” 출력
  2. 실행 중 오류가 났을 경우 [그림 ii]과 같이 처리하시오.
* 오류 로그의 상태와 이름은 빨간색으로 표시하고, 상태는 실패로 출력하시오.
* 오류 발생 시 즉시 “DB 구성 실패” 로그를 출력하고, 종료하시오.
* DB 구성에 실패하면 DB를 지우도록 하시오.
  1. 모든 작업이 끝나면 5초 후에 프로그램이 자동 종료
* 매초마다 종료까지 남은 시간이 표시된다.
* 성공, 실패 유무 상관없이 마지막 작업으로 동작된다.

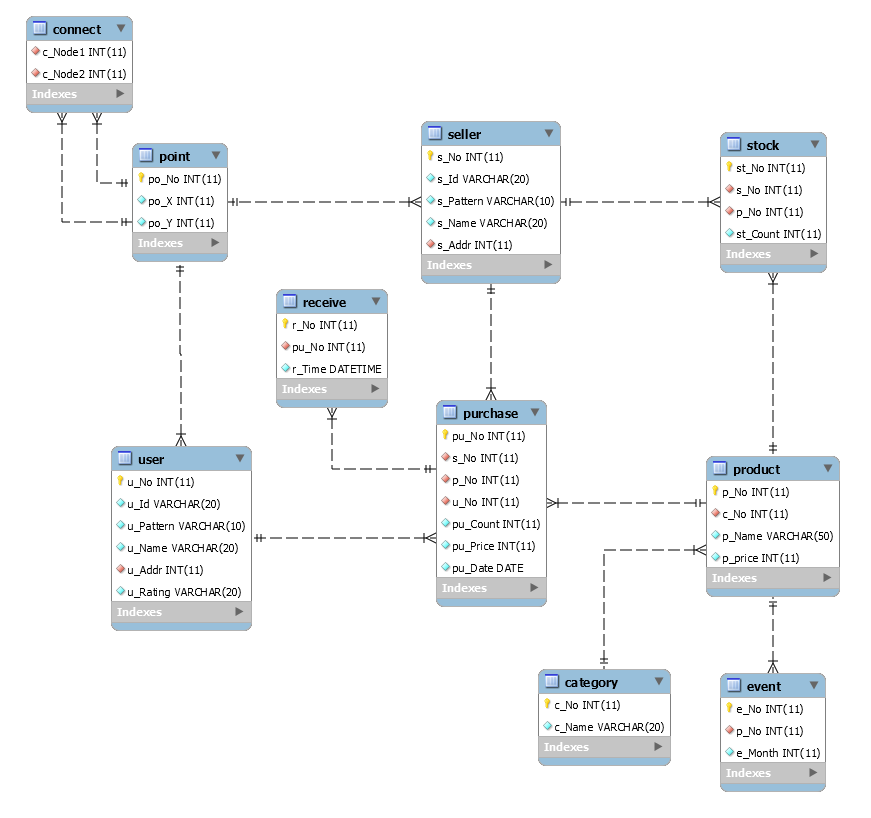


[그림 i] DB 구성 성공



[그림 ii] DB 구성 실패

1. 다음 [그림 iii]과 같이 완성된 DB의 관계도를 작성하여 'Delivery.mwb’로 저장하시오.



[그림 iii] ERD

# [문제2] 다음 조건을 만족하는 ‘Project.jar’를 작성하시오.

1. ‘메인’ 페이지

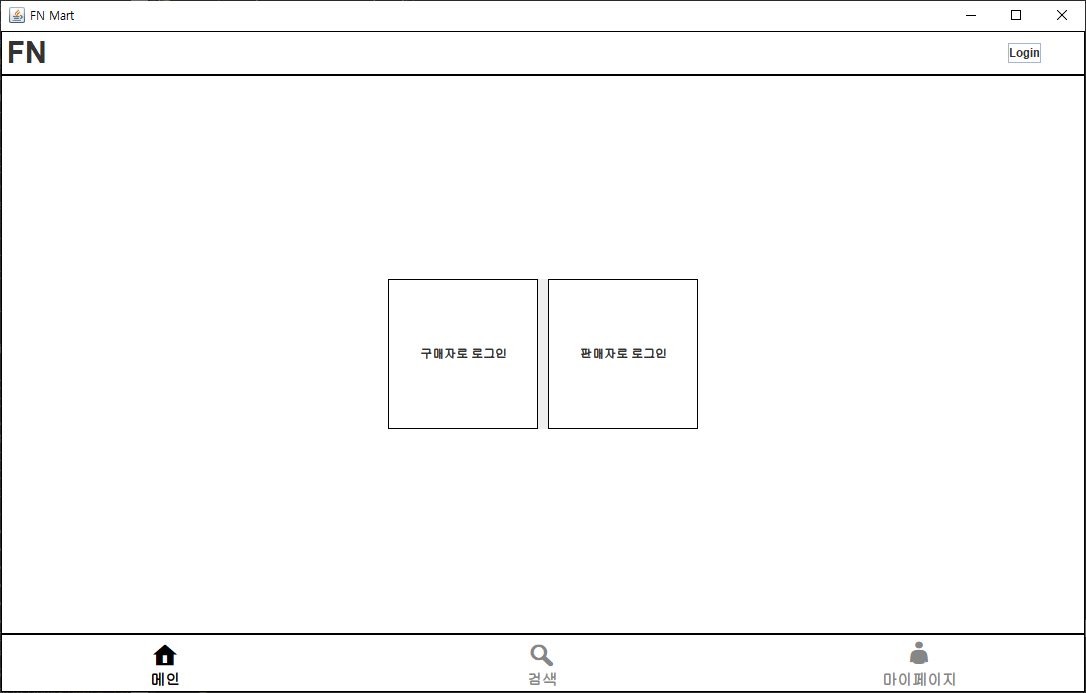


**[그림 1] 메인**

1. ‘Project.jar’를 실행하면 [그림 1] ‘메인’ 폼이 실행되도록 하시오.
2. 상단의 FN을 클릭했을 경우 [그림 1] ‘메인’ 폼으로 이동하시오.
3. [Login] 버튼을 클릭했을 경우 [그림 2] ‘로그인’ 폼으로 이동하시오.
4. 인기 상품/이달의 1+1 행사 상품 영역 내부에서 좌우로 드래그 할 경우 이전/다음 페이지로 넘어가게 하시오.
5. 하단의 [메인] (UTF-8 유니코드 : 127968) / [검색] (UTF-8 유니코드 : 128269) / [마이페이지] (UTF-8 유니코드 : 128100) 버튼을 클릭했을 경우 다음과 같이 제어하시오.
   1. 로그인 하지 않았을 경우 🡪 경고 : “로그인을 하지 않았습니다.”
   2. 로그인 한 경우 🡪 아이콘 버튼을 선택하면 색이 진해지고 해당 폼으로 이동한다.

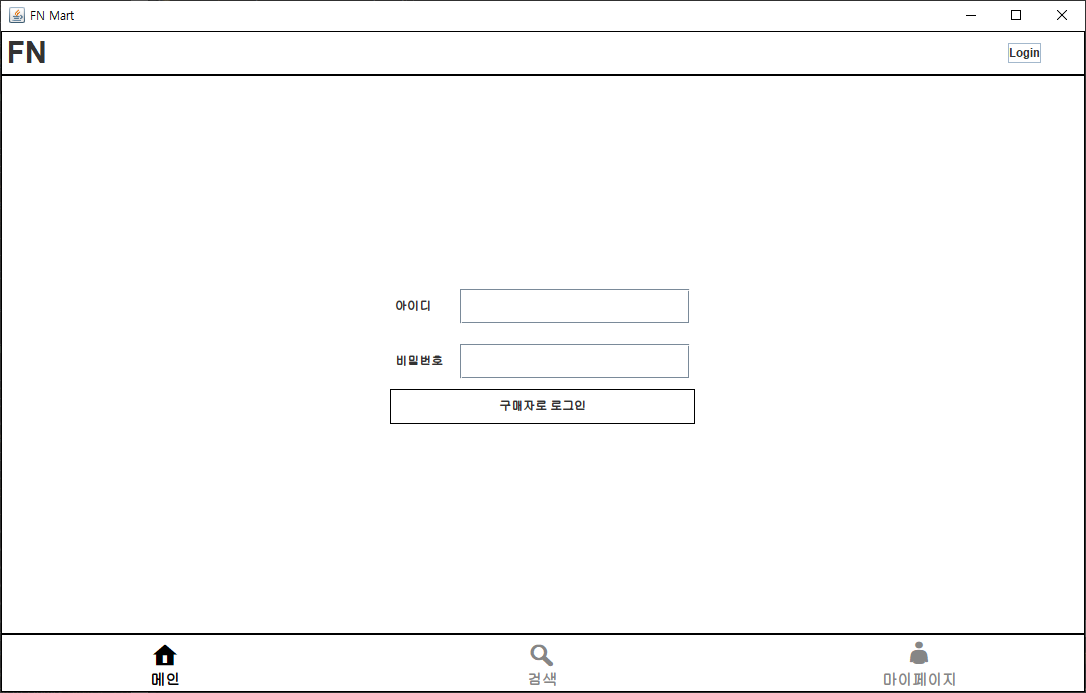
* [메인] 버튼 클릭 시 🡪 [그림 1] ‘메인’ 폼 이동
* [검색] 버튼 클릭 시 🡪 [그림 5] ‘검색’ 폼 이동
* [마이페이지] 버튼 클릭 시 🡪 [그림 7] ‘마이페이지’ 폼 이동

1. X버튼을 클릭하면 프로그램을 종료 하시오.
2. ‘로그인’ 페이지



**[그림 2] 로그인**

1. 로그인 폼을 [그림 2]와 같이 구성하시오.
2. [구매자로 로그인], [판매자로 로그인] 버튼 클릭한 경우 🡪 구매자/판매자 로그인 폼으로 이동하시오.
3. ‘구매자/판매자 로그인’ 페이지



**[그림 3] 구매자/판매자 로그인**

1. 폼을 다음과 같이 구성하시오.
   1. [그림 2] 로그인 폼에서 [구매자로 로그인] 버튼을 클릭한 경우 🡪 버튼 제목 : “구매자로 로그인”
   2. [그림 2] 로그인 폼에서 [판매자로 로그인] 버튼을 클릭한 경우 🡪 버튼 제목 : “판매자로 로그인”
2. 비밀번호는 직접 입력이 불가능하고, 클릭할 경우 [그림 4] ‘패턴’ 폼이 나타나도록 하시오.
3. [구매자/판매자로 로그인] 버튼을 클릭할 경우 다음과 같이 제어하시오.
   1. 빈칸이 있을 경우 🡪 경고 : 빈칸이 있습니다.
   2. 비밀번호가 3자리 이하일 경우 🡪 경고 : “비밀번호는 3자리 이상으로 입력해주세요.”
   3. 아이디 또는 비밀번호가 일치하지 않을 경우 🡪 경고 : “아이디나 비밀번호를 다시 확인해주세요.”
   4. 구매자로 로그인한 경우 🡪 정보 : “OOO 회원님, 환영합니다.” 🡪 [그림 1] ‘메인’ 폼으로 이동
   5. 판매자로 로그인한 경우 🡪 정보 : “OOO 판매자님, 환영합니다.”   
      🡪 [그림 10] ‘판매자 메인’ 폼으로 이동
4. 로그인 성공할 경우 🡪 ‘메인’의 [Login] 버튼의 텍스트를 [Logout]으로 변경하도록 하시오.
   1. [Logout] 버튼을 클릭할 경우 🡪 정보 : “로그아웃 되었습니다.” 🡪 [그림 1] ‘메인’ 폼으로 이동
5. ‘패턴’ 폼

※ 패턴.mp4참조

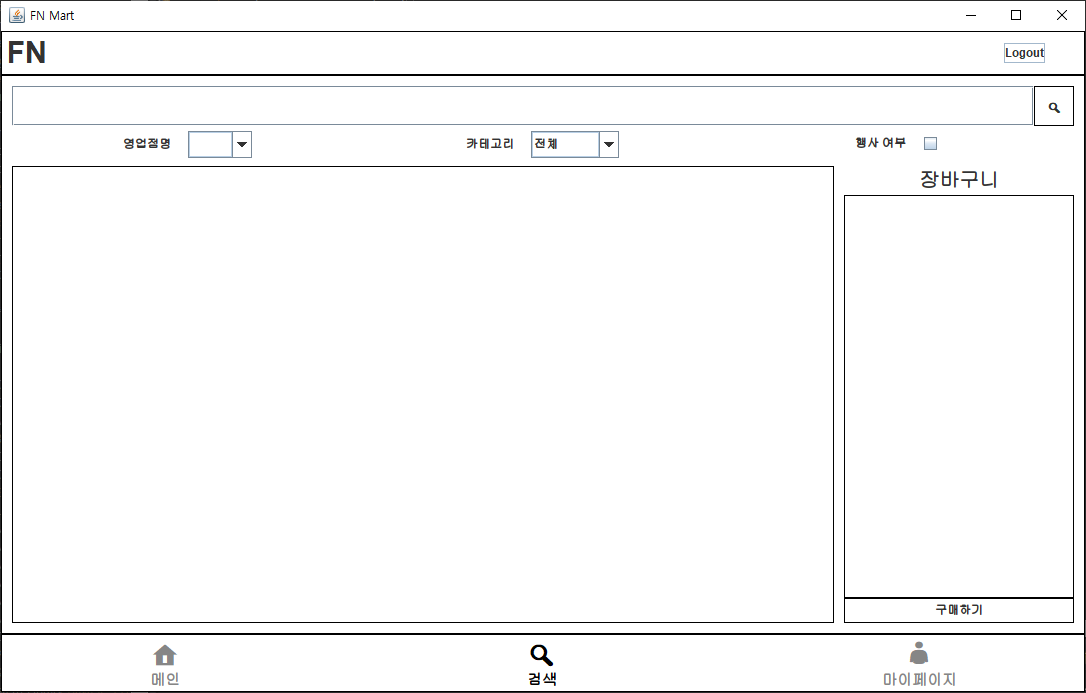
|  |  |
| --- | --- |
|  |  |
| **[그림 4] 패턴** | **[그림 4-1] 2 🡪 6** |

1. 패턴은 위에서부터 123 / 456 / 789 로 구성하시오.
2. 원에서 다른 원으로 드래그 할 경우 [그림 4-1]과 같이 연결되도록 하시오.
   1. 동일한 숫자는 한 번만 선택 되도록 한다.
   2. 원에 접촉하면 해당 숫자가 선택된다.

(예1) 1 🡪 4 🡪 1 의 경우 : “14”

(예2) “17” 을 표현하는 방법 : 1번 원 접촉 🡪 4번 원0을 피함 🡪 7번 원 접촉

1. [확인] 버튼을 클릭할 경우 [그림 3] ‘구매자/판매자 로그인’ 폼의 비밀번호란에는 패턴 폼에서 라인이 연결된 순서대로 숫자가 특수문자 ‘●’ 로 입력되도록 하시오.
2. 패턴.mp4 참조
3. ‘검색’ 페이지



**[그림 5] 검색 초기화면**

1. 초기 화면은 [그림 5] 처럼 나타내시오.
   1. 영업점 콤보박스에는 seller 테이블의 모든 영업점명이 나타난다.
   2. 영업점을 변경하면 물품 목록은 해당 영업점의 물품 목록, 카테고리는 “전체”, 검색란은 빈칸, 행사 여부 체크박스와 장바구니는 초기화된다.
2. [검색] 기능을 [그림 5-1]과 같이 제어하시오.
   1. 상품명, 영업점명, 카테고리, 행사여부가 AND 조건으로 검색된다.

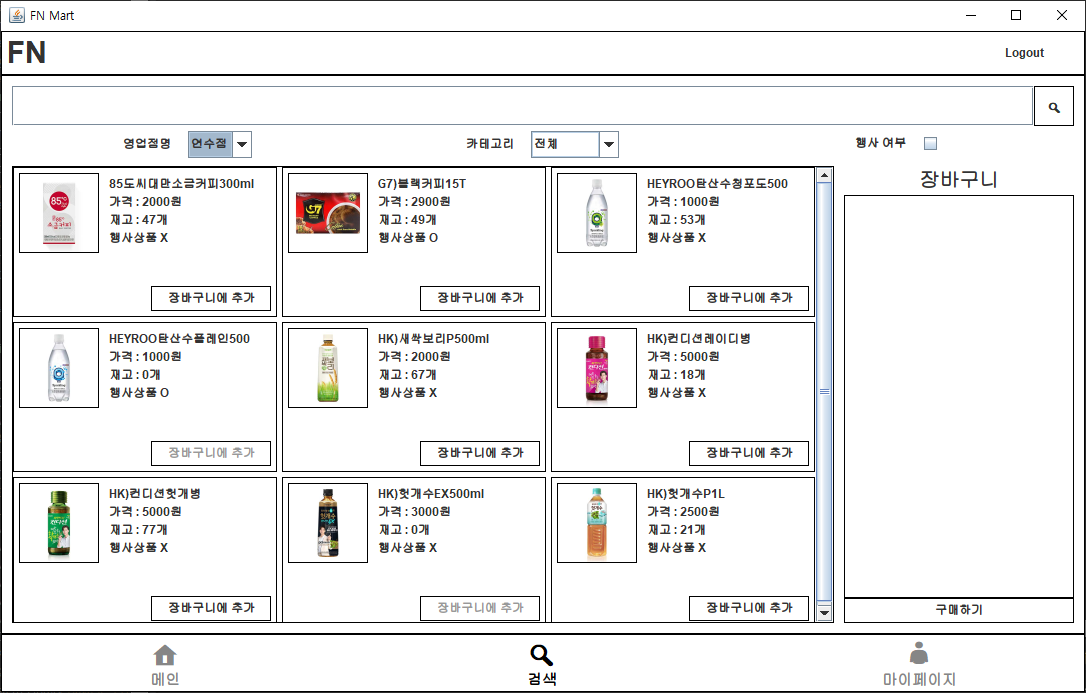
- 상품명 일부를 입력할 경우 🡪 상품명 일부가 포함된 상품만 검색

- 상품명이 빈칸일 경우 🡪 조건에 맞는 모든 상품 검색

- 행사 여부를 체크할 경우 🡪 행사 상품만 검색

- 행사 여부를 체크하지 않을 경우 🡪 행사 여부 상관없이 모든 상품 검색

* 1. 검색 결과는 product 테이블의 최상단 9개만 나타나고, 스크롤을 최대로 내리면 그 다음 상품 9개가 추가되도록 한다.



**[그림 5-1] 검색 기능**

1. 상품 정보는 다음과 같이 [그림 5-2] 와 같이 나타내시오.
   1. 행사 상품일 경우 🡪 ‘O’으로 표시, 행사 상품이 아닐 경우 🡪 ‘X’로 표시
   2. 재고가 0개일 경우 🡪 [장바구니에 추가] 버튼 비활성화

|  |  |
| --- | --- |
|  |  |
| **[그림 5-2] 상품 정보** | |

1. [장바구니에 추가] 버튼을 클릭할 경우 다음과 같이 제어하시오.
   1. 선택한 상품이 장바구니에 추가된다.
   2. 장바구니에 이미 추가 되어있는 상품을 다시 추가할 경우  
      🡪 경고 : “이미 추가된 상품입니다.”
   3. 버튼을 클릭할 경우 🡪 장바구니에서 삭제된다.
   4. 장바구니가 넘칠 경우 [그림 5-3]와 같이 스크롤이 가능하게 한다.

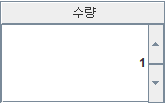
|  |  |
| --- | --- |
|  |  |
| **[그림 5-3] 장바구니** | |

1. [구매하기] 버튼을 클릭할 경우 다음과 같이 제어하시오.
   1. 장바구니에 담은 상품이 없을 경우 🡪 경고 : “장바구니가 비어있습니다.”
   2. 장바구니에 상품이 있을 경우 🡪 [그림 6] ‘구매’ 폼으로 이동한다.
2. 검색.mp4 참조
3. ‘구매’ 페이지



**[그림 6] 구매**

1. 구매 폼을 [그림 6]과 같이 나타내시오.
   1. 장바구니에 담았던 상품 정보를 테이블에 표시한다.
   2. 수량 필드를 제외한 나머지 필드들은 입력이 불가능하도록 한다.
   3. 수량 필드를 더블클릭하면 [그림 6-1]과 같이 스핀 단추가 나타난다.
   4. 금액은 총합 필드를 합산한 결과를 천 단위 구분 기호로 구분하여 표시한다.
   5. 행사 상품인 경우 🡪 수량은 짝수로만 입력된다.

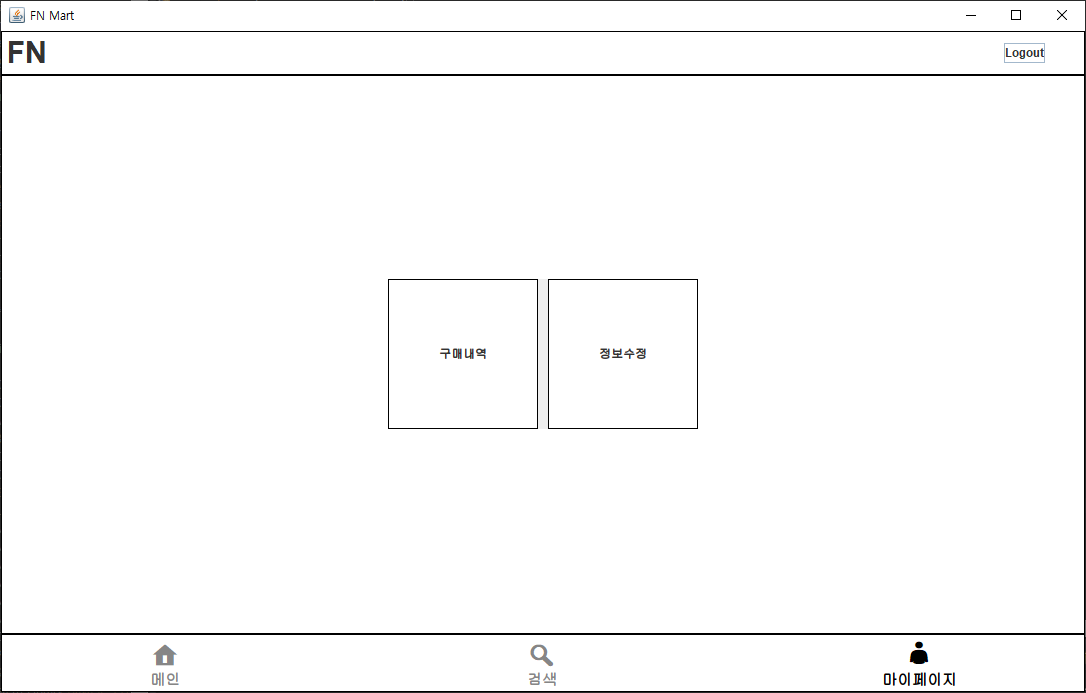


**[그림 6-1] 스핀버튼**

1. 수량 필드의 스핀 버튼을 다음과 같이 제어하시오.
   1. 수량은 1 ~ 재고개수까지 1개씩 증감하되, 행사 상품인 경우 2개씩 증감 처리한다.
   2. 수량을 증감 할 경우 총합과 금액을 자동 계산하되, 행사 상품의 경우 2개 당 1개의 가격으로 계산한다.
   3. 구매\_스핀버튼.mp4 참조
2. [구매하기] 버튼을 클릭할 경우 다음 예시와 같이 제어 하시오

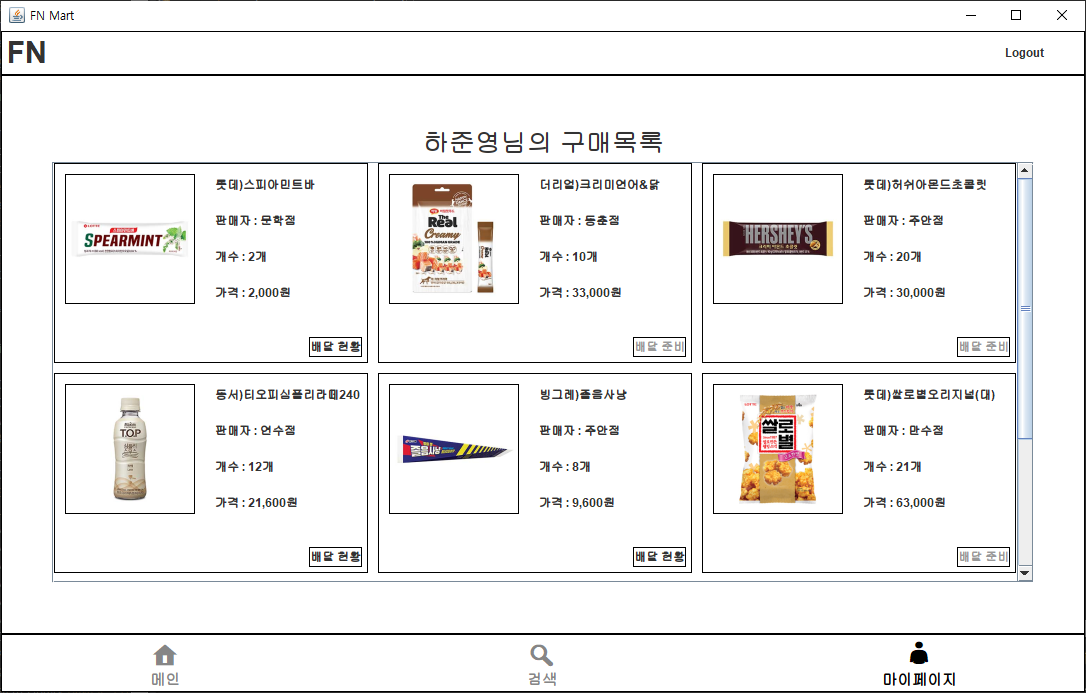
(예시) 정보 : “총 21,000원 결제 완료되었습니다.” 🡪 [그림 1] ‘메인’ 폼으로 이동한다.

1. ‘마이페이지’ 페이지



**[그림 7] 마이페이지**

1. [구매내역] 버튼을 클릭할 경우 [그림 8] ‘구매내역’ 폼으로 이동하시오.
2. [정보수정] 버튼을 클릭할 경우 [그림 9] ‘정보수정’ 폼으로 이동하시오.
3. ‘구매내역’ 페이지



**[그림 8] 구매내역**

1. 로그인한 회원이 구매한 상품 6개가 최초로 나타나고, 스크롤을 최대로 내리면 다음 6개가 [그림 8-1]과 같이 추가되도록 하시오.



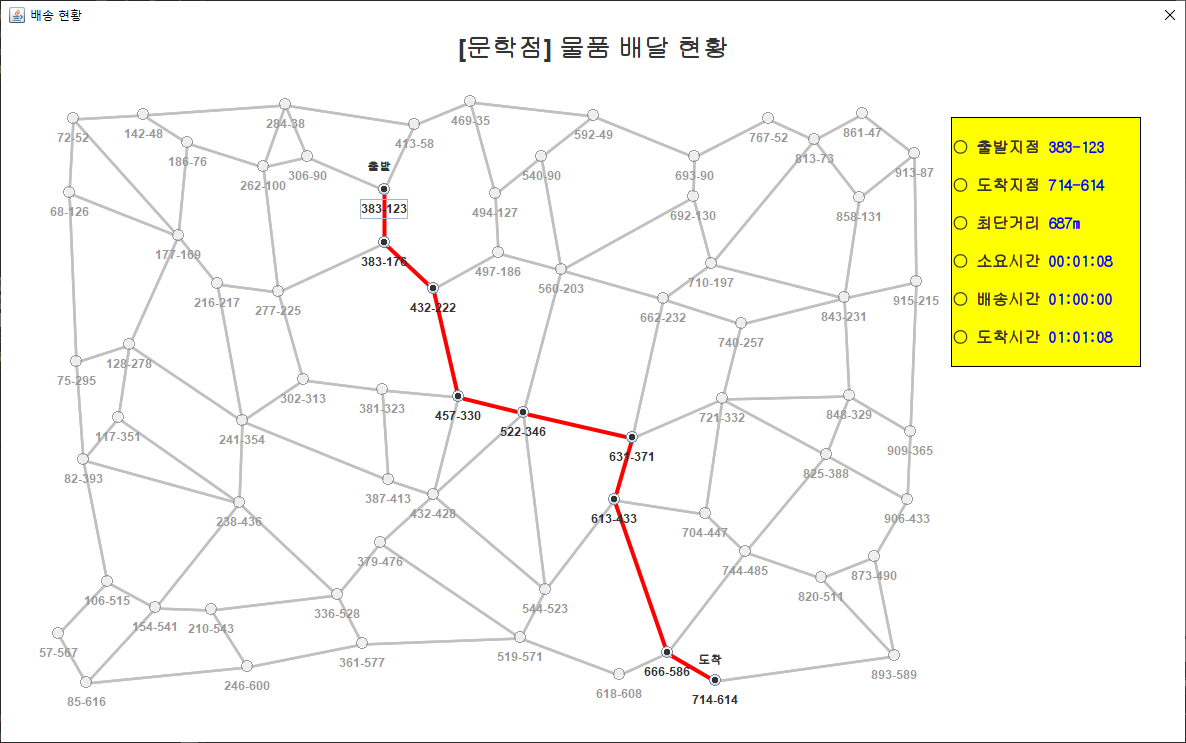
**[그림 8-1]**

1. [배달 현황] 버튼을 다음과 같이 제어하시오.
   1. 배달이 시작되지 않은 경우 “배달 준비”, 버튼은 비활성화한다.
   2. 배달이 시작 된 경우 [그림 10] ‘경로’ 폼이 나타난다.
2. ‘정보수정’ 페이지



**[그림 9] 정보수정**

1. 로그인한 회원 정보가 나타나도록 하시오.
2. [비밀번호] 텍스트박스를 클릭할 경우 [그림 4] ‘패턴’ 폼이 나타나도록 하시오.
3. [주소] 콤보박스는 다른 회원의 주소(u\_Addr)를 제외한 주소(po\_X, po\_Y)가 오름차순으로 나타나도록 하시오.
   1. 다른 회원의 주소를 제외하고 좌표가 오름차순인 것으로 나타나도록 한다.
   2. 초기 화면의 콤보박스에는 현재 회원의 주소가 나타나도록 한다.
4. [수정] 버튼을 클릭할 경우 다음과 같이 제어하시오.
   1. 빈칸이 있을 경우 🡪 경고 : “빈칸이 있습니다.”
   2. 비밀번호가 3자리 이하일 경우 🡪 경고 : “비밀번호는 3자리 이상으로 입력해주세요.”
   3. 조건에 맞을 경우 🡪 정보 : “회원정보가 수정되었습니다.” 🡪 [그림 7] ‘마이페이지’ 폼으로 이동한다.
5. ‘경로’ 폼

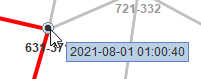


**[그림 10] 경로**

1. 도로 망을 [그림 10]과 같이 생성하시오.
   1. point 테이블의 X, Y 좌표를 이용하여 도로망을 생성한다.
   2. 모든 지점에 라디오 버튼을 표시하고, 지점의 주소를 “OOO-OOO” 형식으로 표시한다.
   3. 배달 경로가 아닌 지점의 라디오 버튼은 모두 비활성화 한다.
   4. 배달 경로에 있는 지점의 라디오 버튼은 활성화 하되, 임의로 선택되지 못하게 한다.
2. 물품 배송 경로를 다음과 같이 제어하시오.
   1. 판매자 지점에서 구매자 지점까지 총 이동거리가 가장 짧은 경로(최단 경로)가 표시되도록 한다.
   2. 연결된 두 지점 간의 거리는 해당 지점의 X, Y 좌표를 이용하여 직선 거리를 구하되, 소수점 이하를 버리고 정수로 계산한다. (단, 계산 결과는 m로 간주한다.)

예1) 좌표 (1, 2)와 좌표 (4, 5) 간의 거리 계산 :

* 1. 배달 경로 상의 모든 지점에 마우스가 over되면 [그림 10-1]와 같이 해당 지점의 도착시간이 표시되도록 한다.



[그림 10-1] 도착시간 표시

1. 배송 구간은 다음과 같이 제어하시오.
   1. 출발 좌표와 도착 좌표에 [출발], [도착] 레이블이 나타나도록 한다.
   2. 구간 진입 전일 경우 경로 색 🡪 빨간색
   3. 구간 배송 중일 경우 경로 색 🡪 [그림 10-2]과 같이 파란색 - 빨간색 점멸
   4. 구간 통과인 경우 경로 색 🡪 [그림 10-3]와 같이 파란색

|  |  |
| --- | --- |
|  |  |
| [그림 10-2] 구간 배송 중 | [그림 10-3] 배송 완료 |

1. 제목은 [구매내역]에서 선택한 판매 지점명이 나타나고, 우측의 배송 정보를 다음과 같이 나타나도록 하시오.
   1. 출발 지점은 판매자 주소, 도착 지점은 구매자 주소 표시한다.
   2. 최단 거리는 출발 지점부터 도착 지점까지 최단 이동 거리로 표시한다.
   3. 소요시간은 10m 당 1초로 환산하되 소수점 이하는 버림한다.
   4. 배송 시간은 배송 시작시간으로 표시한다.
   5. 도착 시간은 배송 시간 + 소요 시간으로 계산한다.
2. 배송 구간에 따른 시간은 현재 시스템 시간으로 비교하도록 하시오.
3. 경로.mp4 참조

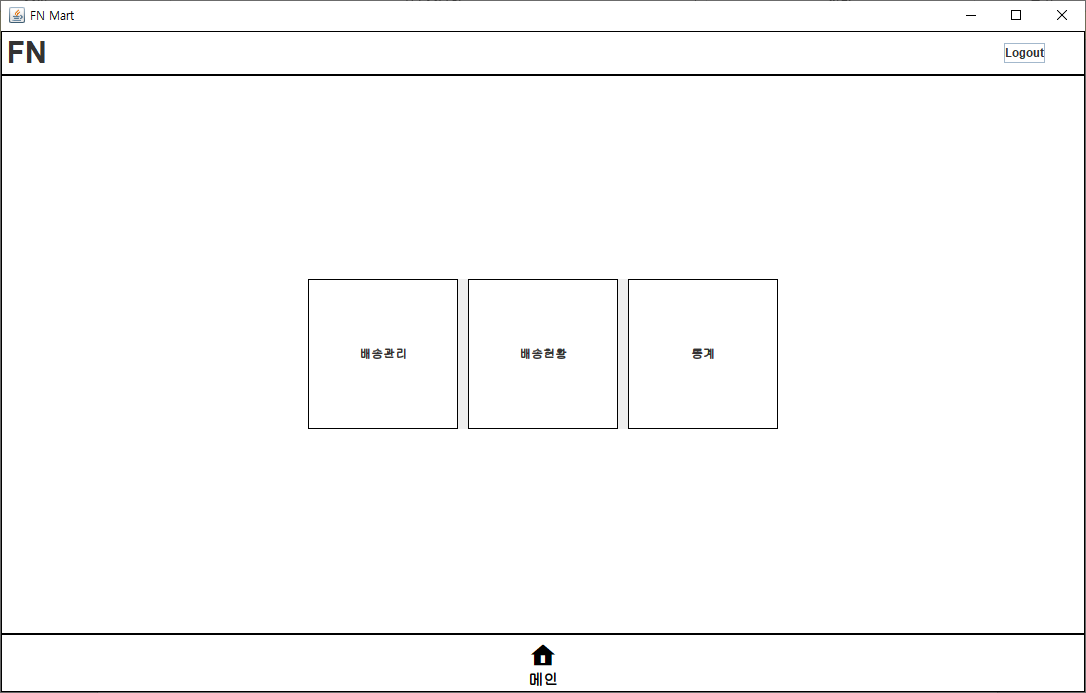
* 경로 테스트 방법

순서1. 구매자로 로그인하여 물품을 구매 처리한다.

순서2. 판매자로 로그인하여 배송관리 페이지에서 구매한 물품의 [배달하기] 버튼을 누른다.

순서3. 구매자로 로그인하여 구매한 물품의 [배달 현황] 버튼을 누른다.

1. ‘판매자 메인’ 페이지



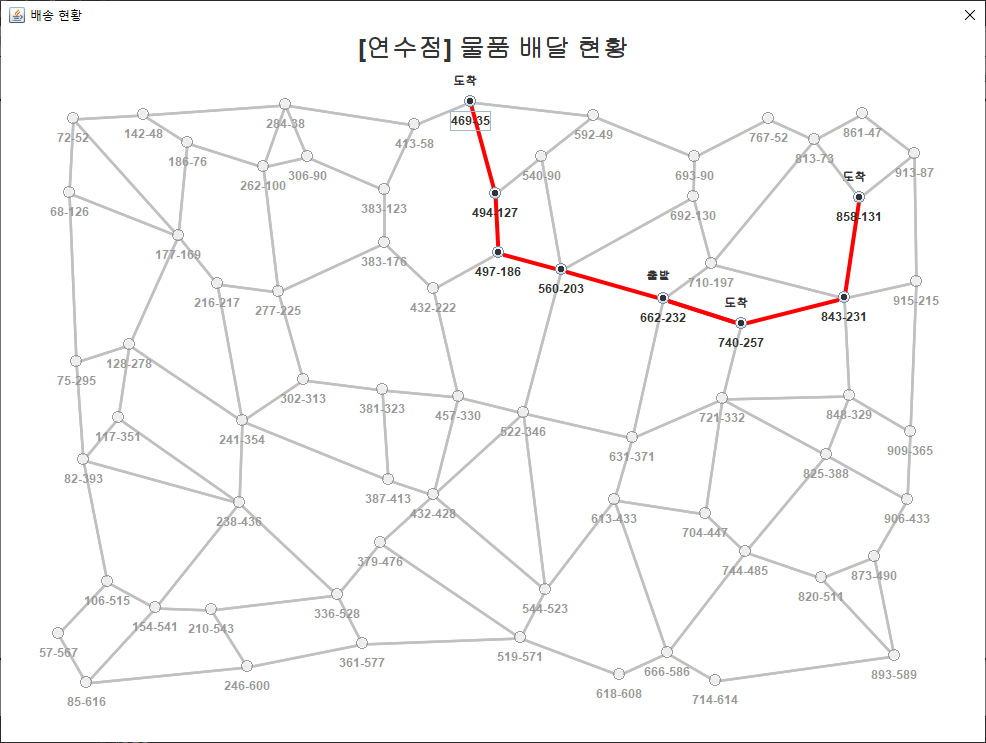
**[그림 10] 판매자 메인**

1. 하단은 [메인] 버튼만 존재하도록 하시오.
2. [배송관리] 버튼을 클릭할 경우 [그림 11] ‘배송 관리’ 폼으로 이동하시오.
3. [배송현황] 버튼을 클릭할 경우 [그림 12] ‘배송 현황’ 폼으로 이동하시오.
4. [통계] 버튼을 클릭할 경우 [그림 13] ‘통계’ 폼으로 이동하시오.
5. ‘배송관리’ 페이지



**[그림 11] 배송관리**

1. 로그인한 판매자가 판매한 상품 중 미 배송 상품만 모두 표시하되, 최초 6개가 먼저 나타나고, 스크롤을 최대로 내리면 다음 6개가 [그림 11]과 같이 추가되도록 하시오.
2. [배달하기] 버튼을 클릭할 경우 다음과 같이 제어하시오.
   1. 정보 : “배송을 시작합니다.”
   2. 현재 상품의 배송 시작 시간이 현재 시스템 날짜/시간으로 등록되도록 DB를 업데이트한다.
   3. 미배송 상품만 표시되도록 화면을 갱신하도록 한다.
3. ‘배송현황’ 폼



**[그림 12] 배송현황**

1. 10. 1)과 같은 조건으로 [그림 12]와 같이 도로망을 생성하시오.
2. 물품 배송 경로를 다음과 같이 제어하시오.
   1. 배송 시작 날짜가 시스템 날짜와 일치하는 모든 물품의 배송현황을 동시에 표시하시오.
   2. 판매자 지점에서 구매자 지점까지 총 이동거리가 가장 짧은 경로(최단 경로)가 표시되도록 하시오.
   3. 연결된 두 지점 간의 거리는 해당 지점의 X, Y 좌표를 이용하여 직선 거리를 구하되, 소수점 이하를 버리고 정수로 계산하시오. (단, 계산 결과는 m로 간주한다.)

예1) 좌표 (1, 2)와 좌표 (4, 5) 간의 거리 계산 :

* 1. 최종 도착 지점에 마우스가 over되면 해당 지점의 도착시간이 표시되도록 하시오.

1. 10. 3)과 같은 조건으로 배송 구간을 제어하시오.
2. 배송현황.mp4 참조

* 배송 현황 테스트 방법

순서1. 배송관리 페이지에서 배송할 상품의 [배달하기] 버튼 3~4개 클릭

순서2. 배송 현황 페이지로 이동

1. ‘통계’ 페이지

|  |  |
| --- | --- |
|  |  |
| [그림 14] 통계 | [그림 14-1] 확대 창 |

1. 폼이 활성화 되면 현재 달력이 나타나도록 하시오.
   1. ◀을 클릭하면 이전 월, ▶을 클릭하면 다음 월이 나타난다.
   2. 현재 달의 일자가 아닌 경우 일자를 회색처리한다.
   3. 일자 별 매출 실적을 해당 일자에 그래프로 표시한다.
2. 각 일자를 클릭하면 [그림 14-1]과 같이 매출 현황이 확대되어 나타나도록 하시오.
   1. 차트 영역 좌측 상단에는 일자를 표시한다.
   2. 차트 영역 우측에는 범례를 표시한다.
   3. 막대 높이는 해당 연도의 일일 최다 판매를 기록한 카테고리의 판매수량 대비 카테고리 별 해당 일 판매수량으로 계산한다.
   4. 판매수량은 데이터 계열 바깥쪽 끝에 표시한다.
   5. [OK] 버튼을 클릭하면 확대 창을 종료한다.