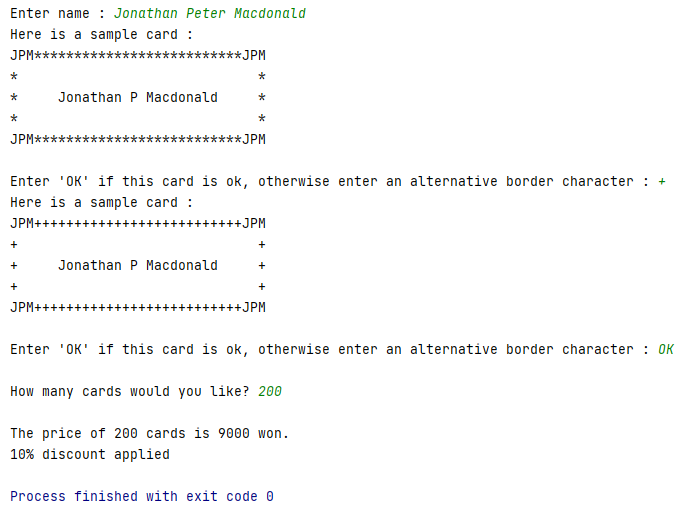
Lab 01



|  |  |
| --- | --- |
| **과목명** | 객체지향 설계와 패턴 |
| **교수님** | 최은만 교수님 |
| **학과** | 컴퓨터공학과 |
| **학번** | 2016112158 |
| **이름** | 김희수 |
| **제출일** | 2021/03/09 |

**실행화면**

**클래스 설명**

MainCard 클래스 : 실행을 담당하는 클래스. 실제 작업을 담당하는 CardInteraction 객체를 생성해서 그 것을 run한다.

CardInteraction 클래스 : 명함에 들어갈 이름, 명함의 테두리, 주문할 명함 개수 등을 입력받아 명함을 담당하는 CardOrder 객체에 전달한다. 또한, 클라이언트가 OK할지 안할지를 결정한다. 명함의 샘플을 출력하는 등의 기능등은 CardOrder객체에 위임한다. 이 클래스는 오직 변수(명함에 들어갈 이름, 테두리, 주문할 명함개수, OK여부)를 입력받고, 요구사항을 디스플레이하는 역할을 한다

CardOrder 클래스 : 명함자체를 관리하는 클래스. 명함에 들어갈 이름, 테두리, 주문된 카드 개수 등을 멤버필드로 받고 그에 대한 getter/setter가 존재한다. 명함을 출력에 관련된 메소드로 getSampleCard(), getTopLine(), getBlankLine(), getLineWithName()이 있다. getTopLine()은 명함의 상단과 하단을 출력한다. 그리고 getBlankLine()은 이름이 포함되지 않은 행을 출력하며, getLineWithName()은 이름이 포함된 행을 출력한다. getSampleCard()에서 앞의 4개의 메소드의 출력결과를 합하여 전체 명함의 샘플을 출력한다. 주문된 명함들의 전체 가격을 계산하기 위한 메소드로 getFinalCost()가 있고, 할인여부를 확인하는 hasDiscount()메소드가 있다.

Name 클래스 : 이름을 담당하는 클래스로써 CardOrder클래스의 구성요소이다. 필드로는 firstName. middleName, lastName이 있으며 middleName이 없는 이름인지를 확인하는 isThereMiddle 필드가 있다. 이 4개의 필드에 대한 getter/setter가 포함되고 이외에도 fistName과 lastName을 합친 결과를 반환하는 getFirstAndLastName() 메소드(예를 들어, John Fitzgerald Kennedy >> John Kennedy 반환), lastName, firstName 형식으로 반환하는 getLastCommaFirst()메소드가 있다(Kennedy, John). 마지막으로 getInits()메소드는 이름의 약자를 반환하는 메소드이다. getInits()메소드는 middleName의 존재여부에 따라 리턴값이 달라진다. 앞에서 봤던 John Fitzgerald Kennedy는 middleName이 존재하므로 JFK가 리턴되지만 이름이 Jane Black이라면 JB가 리턴된다.

MainTest 클래스는 Name의 메소드를 테스트하기 위한 클래스이다.

**클래스 다이어그램**

MainCard, CardInteraction, CardOrder, Name 클래스 관의 관계를 클래스 다이어그램으로 나타내보면 다음과 같다

