안녕하십니까! IT공학전공 19학번 안채린입니다.

저의 최종과제의 주제는 ODIYA COFFEE 키오스크입니다. 키오스크란 요즘 가게에 가면 무인으로 결제하는 기계들이 하나씩 있는데 그 기계들을 통칭하는 말입니다.

ODIYA COFFEE는 커피 프랜차이즈 EDIYA COFFEE에서 영감을 받아 이름을 짓게 되었고 ODIYA COFFEE의 주력 메뉴는 자바칩 프라푸치노입니다.

이 프로그램은 메뉴를 커피, 에이드, 디저트로 나누어서 손님에게 보여준 뒤 메뉴를 누르면 장바구니에 해당 메뉴가 담기게 되고 주문을 마치면 총 금액을 알려주는 기능을 합니다.

계획서에서 추가된 내용으로는 11주차 과제 나만의 계산기를 만들 때 덧셈만 되는 계산기는 너무 간단하고 실용성이 없다고 판단되어서 위에 장바구니 아이콘을 버튼으로 변형하여 버튼을 누르면 가장 최근에 담긴 메뉴를 삭제할 수 있도록 구성하였습니다.

실제 작동을 보기 전에 소스코드에 대해 설명을 하겠습니다.

ODIYA COFFEE는 Mycal.java, MyThread.java, MycalMain.java 이렇게 3가지 코드로 구성이 되는데 첫번째로 Mycal.java부터 설명하겠습니다.

제일 처음에는 전역변수를 선언하고, 메뉴들과 관련된 배열들을 생성하고, 상단의 로고를 위한 객체를 생성합니다.

그리고 프로그램의 오른쪽 장바구니 부분에 들어갈 요소들을 생성하는데 가장 최근 메뉴를 삭제하는 버튼, 선택된 메뉴를 보여주는 스크롤 패널, 그리고 결제창을 실행시키는 결제버튼을 생성합니다. 그리고 각각의 요소를 보더레이아웃을 사용하여 위에서 아래로 차례대로 패널에 붙였습니다.

결제버튼의 구현방식을 먼저 살펴보겠습니다. 결제 버튼이 눌리게 되면 Thread로 구현한 결제창을 start시키게 되고 결제창에는 총 금액과 선택된 메뉴 등을 보여줍니다.

다음으로 가장 최근 메뉴를 삭제하는 버튼을 구현하는 방법을 살펴보겠습니다. 일단 버튼이 눌리게 되면 가장 최근에 추가된 메뉴가 무엇인지 알아내야 하기 때문에 select배열의 가장 마지막 요소인 select[index]의 값을 이용해 삭제할 메뉴의 번호를 찾습니다. 여기서 index는 select에 메뉴가 추가될 때마다 1씩 증가하는 변수로 메뉴가 몇 개 추가되었는지를 나타내게 됩니다. 그 다음으로는 가격배열에서 취소할 메뉴의 가격을 알아내고 전체 가격에서 뺍니다. 그리고 index는 하나 줄이고 선택된 메뉴들을 출력하는 문자열을 업데이트 하고 paintComponent로 구현한 장바구니창은 repaint합니다. Index가 하나 줄었기 때문에 장바구니 창에는 가장 최근의 메뉴가 삭제된 장바구니가 나타나게 됩니다.

그리고 장바구니 창은 java 그래픽으로 구현하였습니다. 일단 메뉴버튼이 눌리면 index가 증가하고 선택된 메뉴의 번호들은 select배열에 저장됩니다. 이를 이용하여 그림을 그리는데 일단 메뉴의 사진은 메뉴 번호와 같이 “bN.png”로 설정을 해두었습니다. 그래서 메뉴번호에 따라 이미지 아이콘과 이미지 객체를 생성하고 각각 icons, img배열에 저장한 뒤 이미지를 panel에 그립니다. 배열에 이미지 아이콘과 이미지 객체를 저장하는 이유는 배열에 저장하지 않고 변수로 지정하면 변수에 있는 값이 계속 바뀌게 되면서 한 종류의 사진 밖에 그려지지 않기 때문입니다. 이 수행을 index만큼 반복하면 장바구니창에 선택된 메뉴들이 그려지게 됩니다. 그리고 취소버튼이 눌리거나 메뉴버튼이 눌릴 때마다 index가 변하고 repaint메소드가 실행되기 때문에 장바구니 창을 이벤트가 발생할 때마다 업데이트 할 수 있습니다.

오른쪽 장바구니 부분에 들어가는 요소에 대한 설명은 마쳤고 다음으로는 메뉴를 보여주고 선택하는 부분을 설명하겠습니다. 일단 메뉴는 커피, 에이드, 디저트로 나누어 보여주어야 하기 때문에 JtabbedPane을 활용하여 구현했습니다. 커피, 에이드, 디저트 패널을 각각 만들고 메뉴 버튼을 붙입니다. 메뉴버튼에서 일어나는 이벤트는 외부 클래스로 다루는데 다음 슬라이드에서 이벤트 처리를 설명하겠습니다.

메뉴 버튼이 눌리게 되면 버튼마다 다른 이벤트 리스너를 등록한 것이 아니기 때문에 어떤 메뉴버튼이 눌러졌는지 알 수가 없습니다. 그래서 제일 처음에 for문을 돌려서 선택된 메뉴가 몇 번인지 찾아야합니다. 그리고 메뉴가 선택되었으므로 index를 하나 증가시키고 Select배열에 추가하고 장바구니는 repaint 해줍니다. 또한 나중에 결과를 출력하기 위해 총 가격과 선택 문자열도 업데이트 해주는 과정까지 이 이벤트 핸들러에서 수행하게 됩니다.

마지막으로 메뉴와 장바구니 그리고 상단로고까지 프레임에 붙이고 프레임 크기, 가시성등을 설정하면 MyCal.java는 끝이 납니다.

다음으로는 결과창을 보여주는 MyThread.java에 대해 설명하겠습니다.

Thread를 생성할 때 MyCal.java에서 이벤트마다 계속 업데이트 해주었던 선택한 메뉴 문자열과 최종금액을 넘겨받게 됩니다. 그리고 Thread가 run될 때 선택한 메뉴의 문자열은 textarea를 이용하여 출력해주고 최종금액은 JLabel을 이용하여 출력합니다. 그리고 결제완료와 결제취소 버튼을 만들고 frame에 붙여줍니다. 결제완료 버튼을 누르면 최종가격이 나타나던 라벨에 “찾아주셔서 감사합니다”라는 문자열이 출력되게 되고 결제취소 버튼을 누르면 결제창의 visible을 false로 바꾸어 결제창이 보이지 않게 만들어주었습니다.

마지막 MyCalMain.java는 Mycal을 생성해주는 메인함수입니다.

코드설명은 마치고 다음으로는 실행결과를 보도록 하겠습니다. 첫 실행화면은 다음과 같습니다.

메뉴 창 상단의 버튼을 누르게 되면 각각의 종류의 메뉴들을 보여줍니다.

이 상태에서 메뉴버튼을 누르게 되면 오른쪽 장바구니창에 선택된 메뉴들이 나타나게 되고 메뉴를 많이 담았다면 스크롤을 내려 선택한 메뉴들을 확인할 수 있습니다.

카트모양의 버튼을 클릭하면 가장 최근에 추가한 메뉴를 삭제할 수 있습니다.

메뉴를 모두 선택하고 난 후에 결제버튼을 누르면 선택한 메뉴와 총 가격을 나누어 보여줍니다.

이 결제창에서 결제취소를 누르면 결제창이 사라지게 되고 결제완료를 누르면 “찾아주셔서 감사합니다.”라는 문구가 나타납니다.

이렇게 ODIYA COFFEE 키오스크를 구현해 보았습니다. 사실 아쉬움도 남는 결과물인데 가장 아쉬운 점은 메뉴를 삭제할 때 가장 최근의 메뉴만을 삭제할 수 있도록 만든 것입니다. 다음에 또 기회가 주어진다면 장바구니창을 조금 더 변형하여 클릭한 메뉴를 삭제할 수 있도록 구현해보고 싶습니다.

그래도 거의 원하던 결과물을 얻게 된 프로젝트여서 힘들기도 했지만 뿌듯한 마음이 더 많이 남는 것 같습니다. 이상으로 모든 발표를 마치도록 하겠습니다. 긴 발표를 들어주셔서 감사합니다.