**모바일 프로그래밍**

**프로그래밍 과제 #3**

**제출일 : 2016. 11. 10.**

**학과 : 컴퓨터공학과**

**학번 : 2014722057**

**이름 : 김진아**

**과제결과요약**

|  |  |  |  |
| --- | --- | --- | --- |
|  | 결과 | 비고 | 자체평가 |
| 1번 | 검증완료 |  | 3점 |
| 2번 | 검증완료 |  | 3점 |
| 3번 | 검증완료 |  | 3점 |
| 4번 | 검증완료 |  | 3점 |
| 5번 | 검증완료 |  | 3점 |
| 6번 | 검증완료 |  | 3점 |

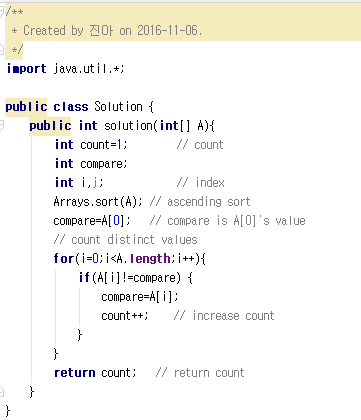
\* 결과 – “검증완료, 디버깅필요, 미구현” 중 하나를 기재할 것.

\* 비고 – 구체적인 사유를 요약해서 기재할 것. (위의 예시 참고)

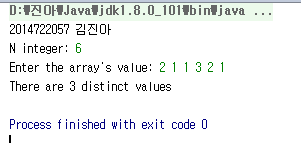
\* 자체평가 기준 - 3점: 검증 완료, 2점: 일부 버그 존재, 0점: 컴파일에러나 미구현

|  |  |  |
| --- | --- | --- |
| 1번 | 결과: 검증 완료 | 자체평가 점수: 3점 |

* 코드



* 실행 결과

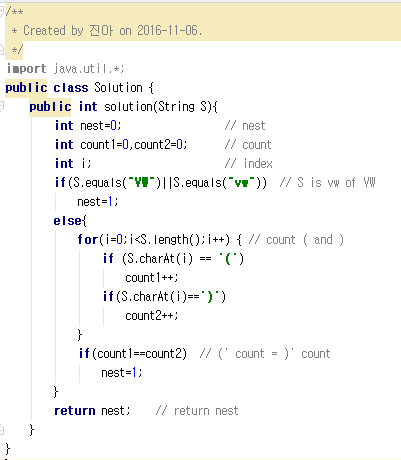
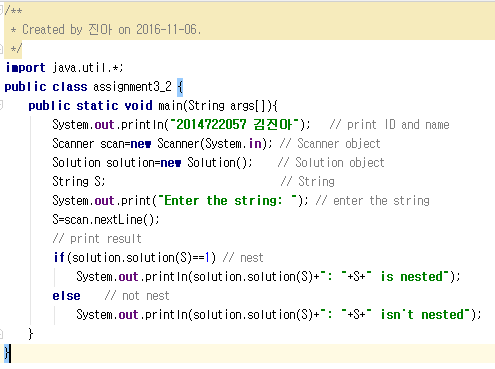


그 외 필요사항 자유롭게 기술

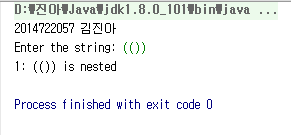
1번은 0부터 시작하는 배열에 몇 개 다른 값이 저장되어 있는지 세는 문제이다. 이를 구현하기 위해 main함수에서 Scanner object를 선언해 nextInt() 함수를 사용해 N을 입력 받아 길이가 N인 배열을 만들었다. 그 후 배열에 값들을 저장하기 위해 nextInt() 함수를 이용했다. 그 다음 Solution class object를 만들어 solution()함수를 불러왔다. 이를 통해 배열에 몇 개의 다른 값이 저장되어 있는지 return 받아 결과를 출력했다. Solution class에 선언된 solution함수에서 배열을 오름차순으로 정렬하기 위해 Arrays.sort()함수를 사용했다. 그 후 for문과 if문을 사용해 몇 개의 다른 값이 저장되어 있는지 셌다. 그 다음 센 값을 return했다.

|  |  |  |
| --- | --- | --- |
| 2번 | 결과: 검증 완료 | 자체평가 점수: 3점 |

* 코드



* 실행 결과

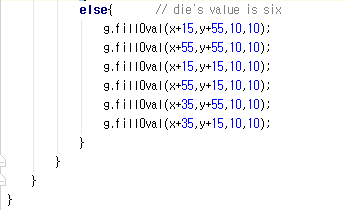
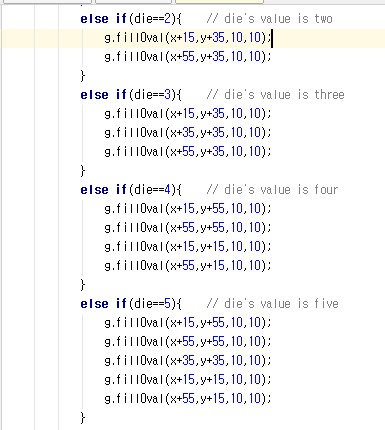
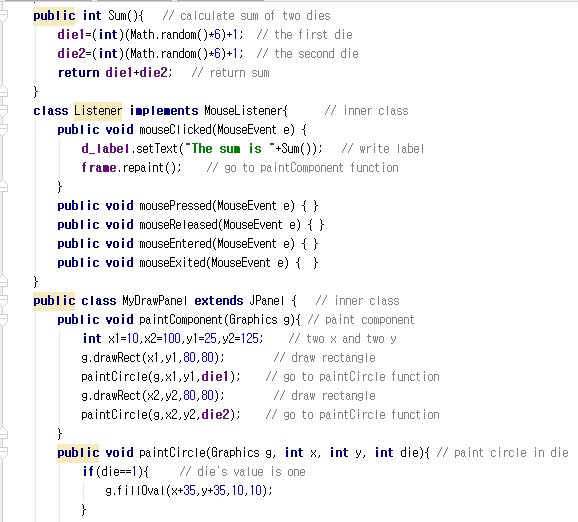
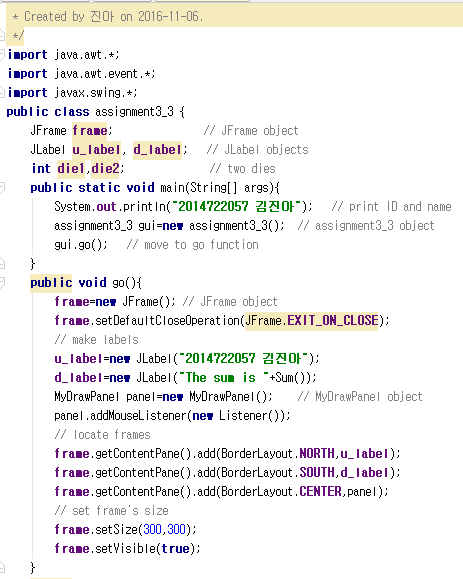


그 외 필요사항 자유롭게 기술

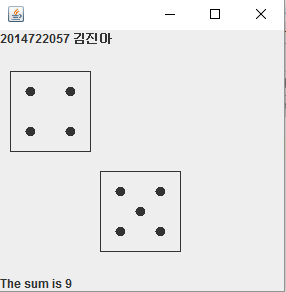
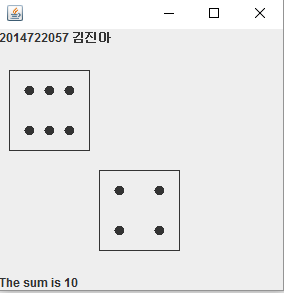
2번은 문자열이 nest인지 아닌지 확인하는 program을 만드는 것이다. main함수에서 Scanner object를 선언해 문자열 S을 nextLine함수를 통해 입력 받도록 했다. 그리고 Solution object를 선언해 Solution class의 solution함수를 불러올 수 있도록 했다. solution함수의 반환 값에 따라 결과를 출력시켰다. Solution class의 solution함수에서 문자열 S가 vw또는 VW일 때 1을 반환하도록 했다. ‘(‘와 ‘)’가 얼마나 포함되었는지 센 후 둘의 개수가 같을 경우 1을 반환하도록 했다. 그 외일 때 0을 반환한다.

|  |  |  |
| --- | --- | --- |
| 3번 | 결과: 검증 완료 | 자체평가 점수: 3점 |

* 코드



* 실행 결과

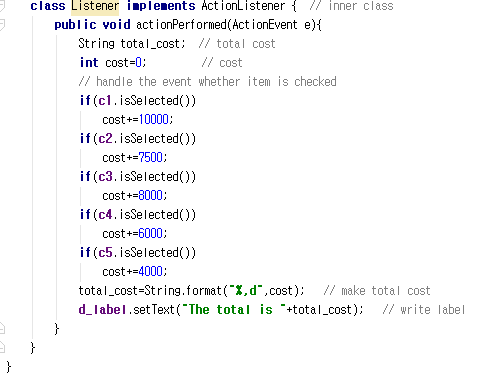
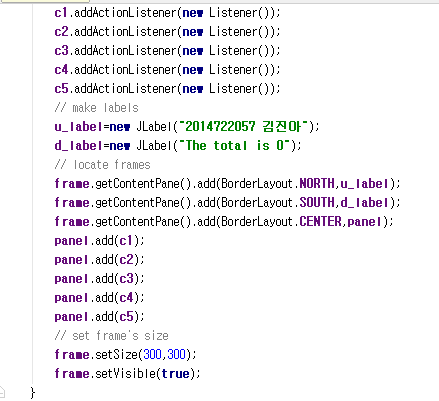
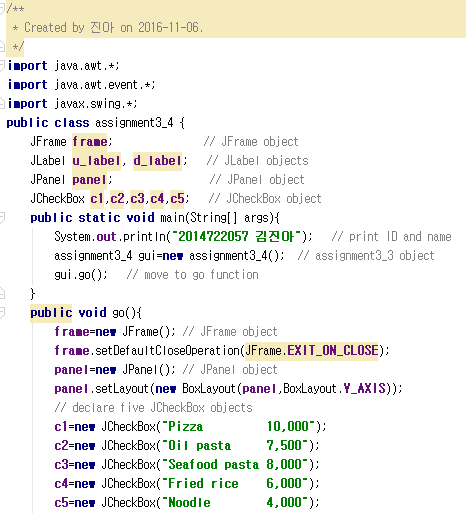


그 외 필요사항 자유롭게 기술

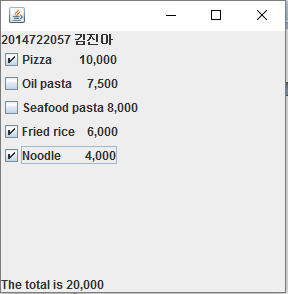
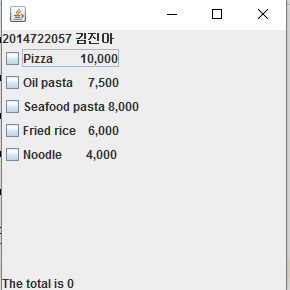
3번은 GUI을 사용해 2개의 주사위의 모습을 보여주고 주사위 결과의 합을 아래에 출력하도록 설계한다. 이때 위에 학번과 이름을 출력하고 panel을 누를 때마다 주사위의 값이 바뀌도록 만든다. frame, label을 만들기 위해 JFrame, JLabel object를 선언했다. 또한 두 개의 주사위 값을 선언해줬다. main함수에서 go함수로 가기 위해 assignment3\_3 object를 선언해줬다. go함수에서 panel이 눌렸을 때 event를 처리해주기 위해 addMouseListener함수를 사용했다. 두 개의 label에 각각 해당하는 내용을 선언해줬다. 두 개의 label과 panel을 각각 북, 남, 가운데에 위치하도록 했다. frame의 size를 300x300로 해줬다. Sum함수에서 두 개의 die의 값을 결정해주고 두 주사위 결과의 합을 반환해줬다. inner class인 Listener class는 event를 발생했을 때 event를 처리해주기 위해 implements MouseListener을 해줬다. mouseClicked함수에서 event가 발생하면 아래에 있는 label의 결과를 바꾸기 위해 Sum함수를 호출해 이에 맞게 주사위 합의 내용을 출력해줬다. 그리고 주사위의 모양을 바꾸기 위해 repaint함수를 사용했다. repaint가 발생하면 inner class인 MyDrawPanel class의 paintComponent함수로 이동한다. 여기서 주사위의 값에 맞게 주사위를 그려줬다.

|  |  |  |
| --- | --- | --- |
| 4번 | 결과: 검증 완료 | 자체평가 점수: 3점 |

* 코드



* 실행 결과

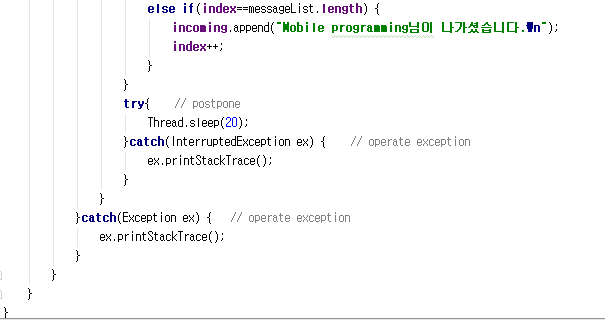
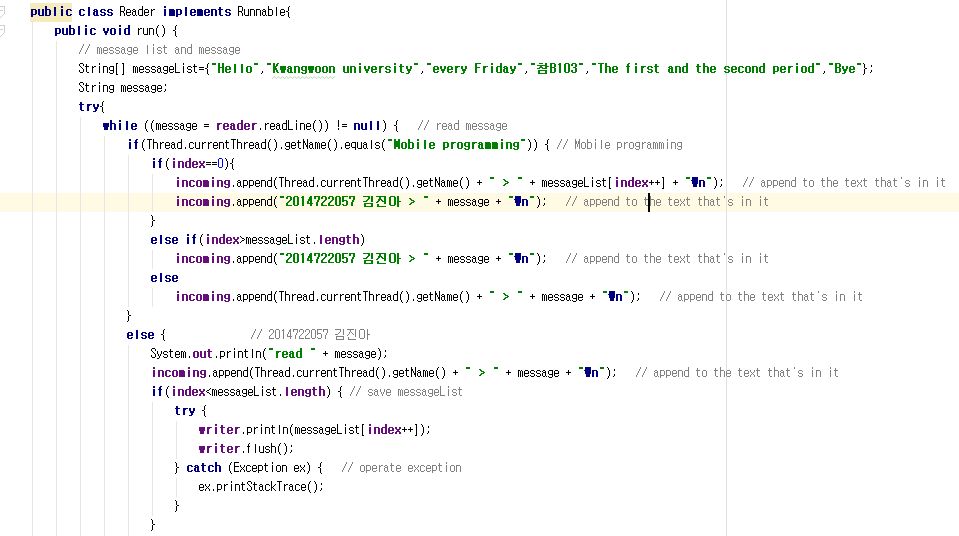
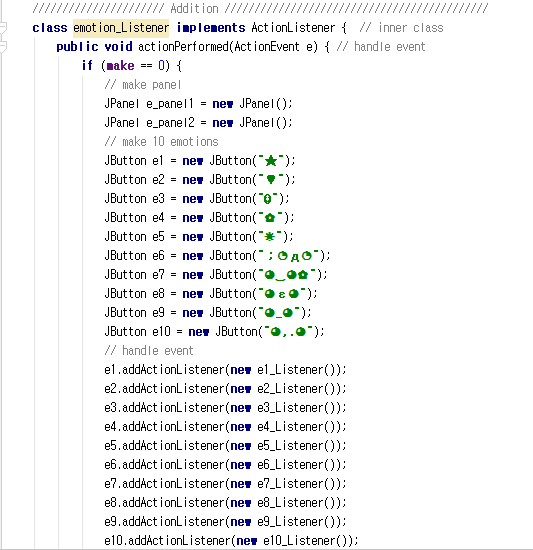
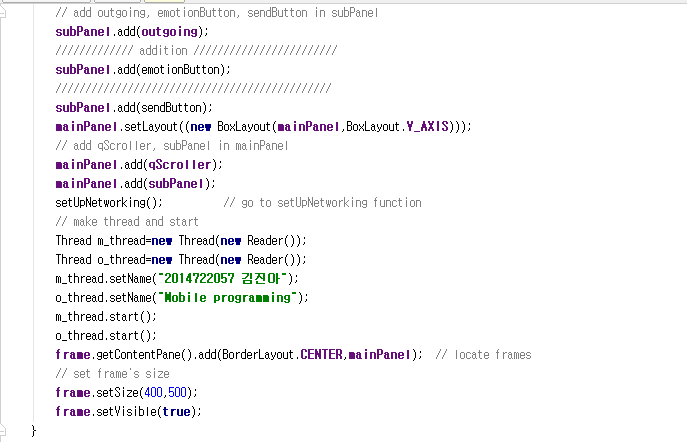
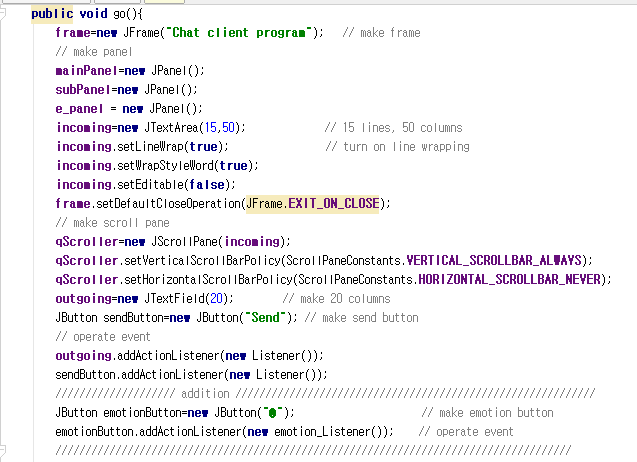
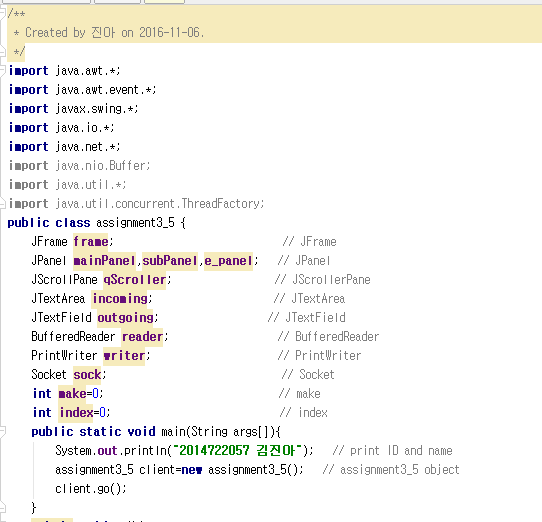
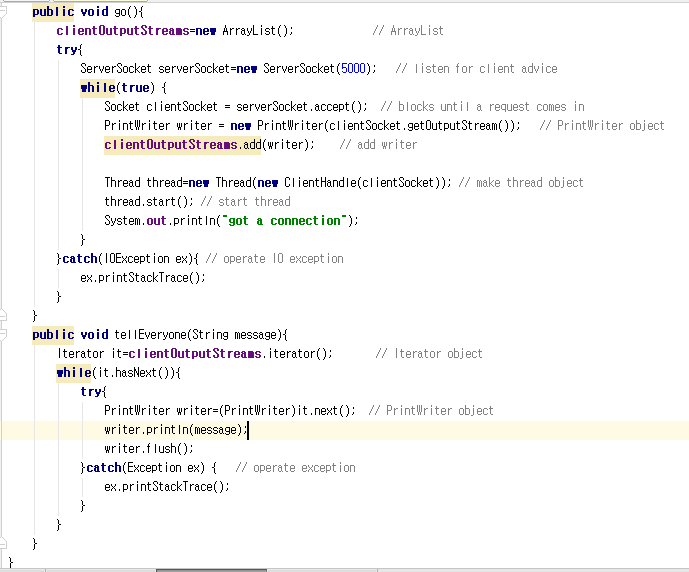
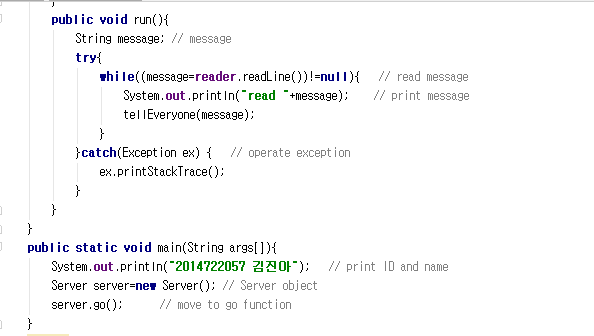
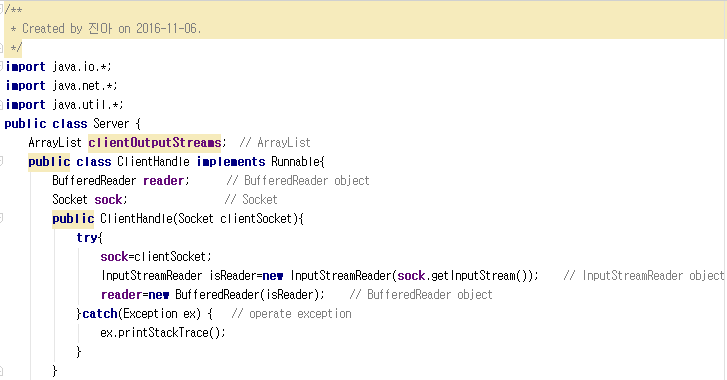


그 외 필요사항 자유롭게 기술

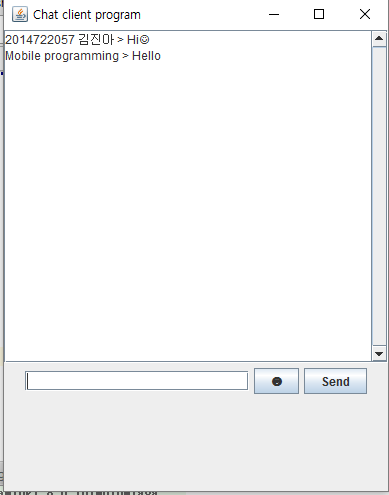
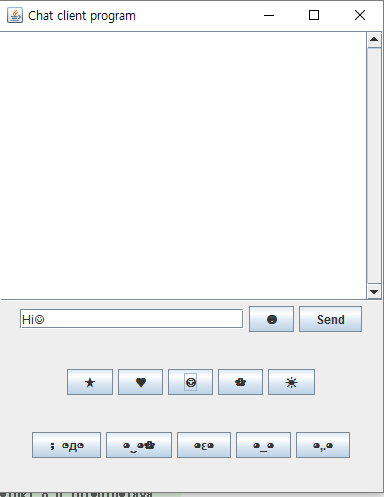
4번은 GUI 사용해 음식을 주문하는 프로그램을 만드는 것이다. 이때 먹고 싶은 음식을 선택하면 총 가격을 계산되어 아래에 보여주도록 한다. 또한 위에 학번과 이름을 출력한다. frame, label, panel, check box를 만들기 위해 JFrame, JLabel, JPanel, JCheckBox object을 선언했다. main함수에서 go함수로 가기 위해 assignment3\_4 object를 만들었다. go함수에서 5개의 checkbox object의 내용을 선언해주었으며 checkbox가 눌렸을 때 event를 다루기 위해 addActionListener함수를 사용했다. 그리고 두 label에 해당하는 내용을 선언해줬다. 그 다음 두 label과 panel의 위치를 각각 북, 남, 가운데로 해줬다. panel에 5개의 checkbox object를 추가해줬다. frame의 size를 300x300로 만들어줬다. inner class인 Listener class를 만들었는데 event가 발생했을 때 처리하는 class이므로 ActionListener을 implements해줬다. event가 발생하면 actionPerformed 함수가 실행된다. 여기서는 5개의 checkbox가 check됐는지 isSelected함수를 이용해 확인한 후 금액을 계산해준다. 천 단위에 콤마를 넣기 위해 String.format함수를 사용했다.

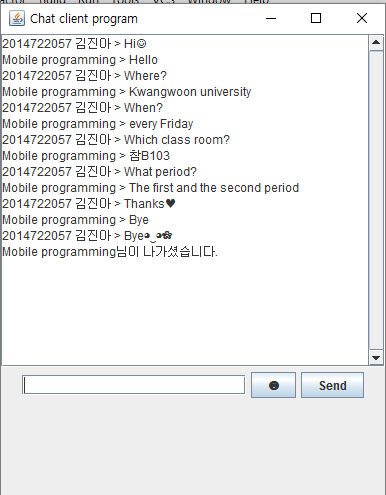
|  |  |  |
| --- | --- | --- |
| 5번 | 결과: 검증 완료 | 자체평가 점수: 3점 |

* 코드



* 실행 결과



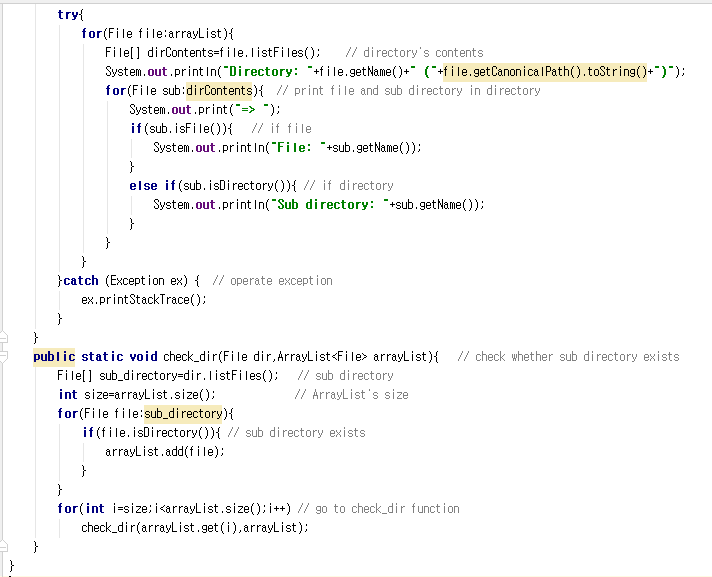
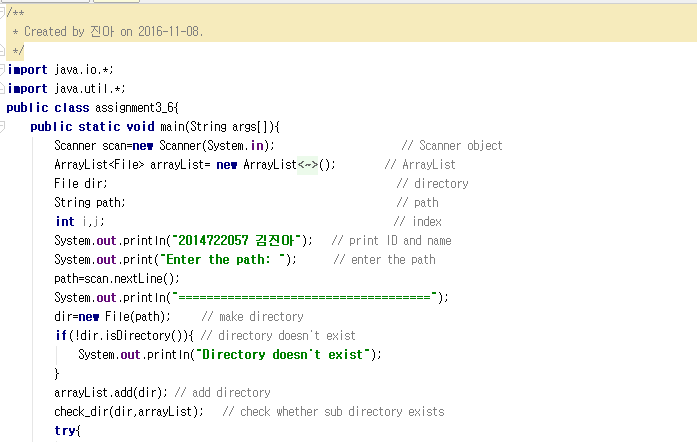


그 외 필요사항 자유롭게 기술

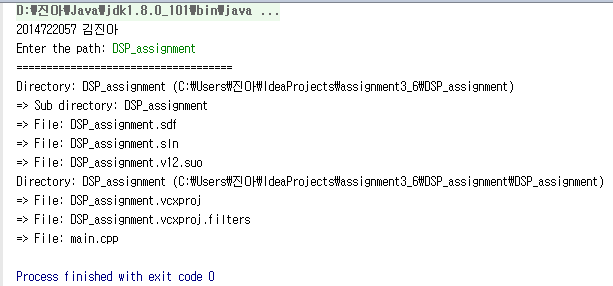
5번은 server와 client가 chat하는 program을 구현하는 것이다. 이때 책에 있는 code을 활용해서 구현한다. client는 GUI를 제공해야 하며 multithread방식이어야 한다. 책에 있는 code에서 text부분에 입력을 했을 때 send 버튼을 누를 때뿐만 아니라 enter칠 때도 event가 발생해 field에 써지도록 했다. 또한 10개의 이모티콘을 쓸 수 있도록 구현했다. 이를 위해 JButton, 3개의 JPanel, 11개의 event처리하는 class와 함수를 생성했다. 첫 번째 class에서는 이모티콘 button을 눌렀을 때 이모티콘이 있는 button 10개를 panel에 만들어 frame에 집어넣었다. 이때 panel이 보이도록 setVisible(true)를 사용했다. 또 다시 이모티콘 button을 누르면 10개의 이모티콘 button을 다시 생성할 필요가 없으므로 panel만 보이도록 했다. 그리고 각각의 10개의 이모티콘을 누를 때 text에 써지도록 나머지 10개의 event처리 class와 함수에 알맞게 써줬다. send button을 누르거나 enter을 치면 이모티콘 panel이 보일 필요가 없으므로 setVisible(false) 해줬다. 두 client와의 대화를 만들기 위해 run함수에서 상대 client의 message List를 저장했다. 그 후에 server로부터 message의 내용을 읽어올 때 상대일 때 경우를 나눠 생각했다. 첫 번째 thread가 상대일 때에는 message List에 있는 내용을 출력한 뒤 server에 저장된 message를 출력했다. 그 외의 경우에서는 message 내용을 출력했다. thread가 자신일 때에는 message를 출력한 뒤 message List에 있는 내용을 출력했다. 이런 식으로 상대방과 대화할 수 있는 program을 만들었다.

|  |  |  |
| --- | --- | --- |
| 6번 | 결과: 검증 완료 | 자체평가 점수: 3점 |

* 코드



* 실행 결과



그 외 필요사항 자유롭게 기술

6번은 특정 directory에 있는 file과 directory들을 출력하는 program을 만드는 것이다. 이때 sub directory에 있는 file과 directory도 출력하도록 구현한다. main함수에서 directory를 저장하기 위해 ArrayList object를 선언했다. 또한 특정 directory을 입력 받기 위해 Scanner object를 선언한 뒤 nextLine함수를 사용해 입력 받았다. directory는 file이므로 type이 File형이다. isDirectory를 사용해 입력 받은 것이 directory인지 확인한다. directory이며 ArrayList에 add함수를 사용해 삽입한다. 그 뒤 check\_dir함수로 가 모든 directory를 저장한다. ArrayList에 있는 directory의 file과 sub directory를 출력했다. 이때 File형 배열을 만들어 listFiles함수를 사용해 directory 안에 존재하는 file들을 저장했다. isFile함수와 isDirectory함수를 통해 file과 directory에 맞게 출력했다. check\_dir함수에서 인자로 받은 File형 변수에 있는 file들을 File형 배열에 listFiles함수를 통해 저장했다. 그 뒤 ArrayList의 크기를 size함수를 통해 저장한다. sub directory가 존재하는지 isDirectory함수를 통해 판별한 뒤 존재하면 ArrayList에 add함수를 통해 저장한다. 이 과정에서 저장한 directory의 sub directory가 있는지 확인하기 위해 for문을 돌려 기존의 ArrayList의 크기부터 새로운 ArrayList의 크기만큼 check\_dir함수로 간다.