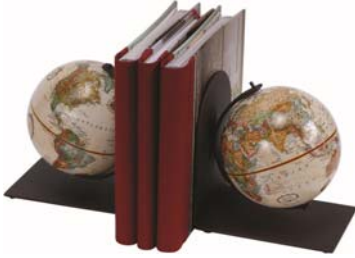


2. GUI와 데이터: 데이터 입력 위젯 (8장)



순천향대학교 컴퓨터공학과
이 상 정

GUI 설계기법

학습 내용

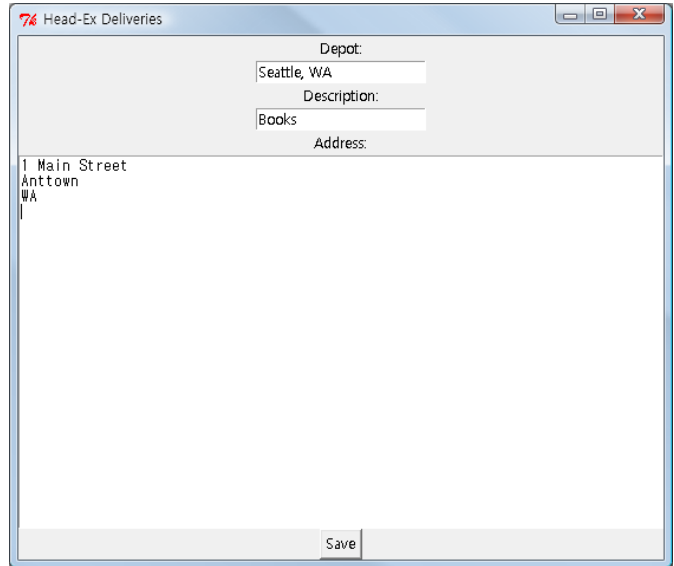
- GUI는 이벤트뿐만 아니라 데이터도 입력
- 입력 위젯 (Entry Widget)
 - 한 라인의 텍스트 입력
- 텍스트 위젯 (Text Widget)
 - 여러 라인의 긴 텍스트 입력
- 라디오 버튼 위젯 (Radio Button Widget)
 - 그룹 중 하나의 항목을 선택
- 옵션 메뉴 위젯 (Option Menu Widget)
 - 많은 항목 중 하나를 선택
 - 드롭다운 리스트 상자(dropdown list box)



헤덱스 배달회사 예

헤덱스(Head-Ex) 배달 예약 시스템

- 사용자가 물건에 대한 설명 (Description), 주소(Address), 창고(Depot) 등 **배달 정보를 입력**
=> 이벤트를 생성하는 것 외에 **데이터를 생성**
- 버튼을 누르면 **deliveries.txt** 파일에 저장



입력 위젯 (Entry Widget)

한 라인의 텍스트 입력

입력 위젯 객체 생성: `Entry()`

- `my_entry_field = Entry(app)`

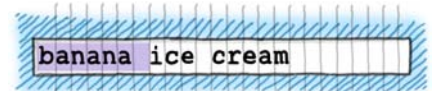


입력 위젯에 입력된 필드의 데이터 읽기: `get()` 메소드

- `my_entry_field.get()`

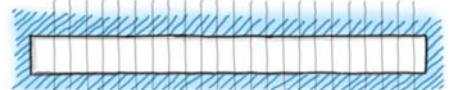
입력 위젯의 필드에 텍스트 추가: `insert()` 메소드

- `my_entry_field.insert(0, "banana ")`
• 0열에 "banana " 삽입



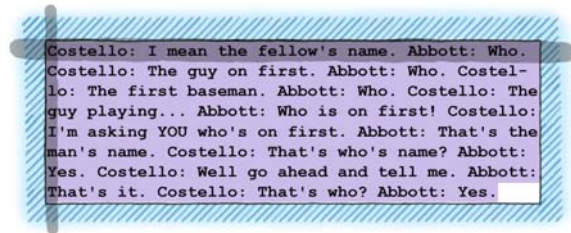
입력 위젯의 필드 내용 지우기: `delete()` 메소드

- `my_entry_field.delete(0, END)`
• 0열에서 END까지 지우기, 전체 내용 지우기
- END는 마지막 글자의 행을 표시



텍스트 위젯 (Text Widget)

- ❑ 여러 라인의 긴 텍스트 입력
- ❑ 텍스트 위젯 객체 생성: `Text()`
 - `my_text_field = Text(app)`
- ❑ 텍스트 위젯에 입력된 필드의 데이터 읽기: `get()` 메소드
 - `my_text_field.get("1.0", END)`
 - 1행 0열(1.0)에서 마지막 글자까지 읽기, 전체 텍스트 읽기
- ❑ 텍스트 위젯의 필드에 텍스트 추가: `insert()` 메소드
 - `my_text_field.insert("1.0", "Some text")`
 - 필드의 제일 앞 부분(1.0)에 "Some text" 삽입
- ❑ 텍스트 위젯의 필드 내용 지우기: `delete()` 메소드
 - `my_text_field.delete("1.0", END)`
 - 제일 앞 부분(1.0)에서 END까지 지우기, 전체 내용 지우기



헤데스 배달회사 코드 (1)

```

from tkinter import *

def save_data():
    fileD = open("deliveries.txt", "a") # 파일 오픈, 파일 뒤에 텍스트 (append)
    fileD.write("Depot:\n")
    fileD.write("%s\n" % depot.get())
    fileD.write("Description:\n")
    fileD.write("%s\n" % description.get())
    fileD.write("Address:\n")
    fileD.write("%s\n" % address.get("1.0", END))
    depot.delete(0, END)
    description.delete(0, END)
    address.delete("1.0", END)

app = Tk()
app.title('Head-Ex Deliveries')

```

헤데스 배달회사 코드 (2)

```
Label(app, text = "Depot:").pack()
depot = Entry(app)
depot.pack()

Label(app, text = "Description:").pack()
description = Entry(app)
description.pack()

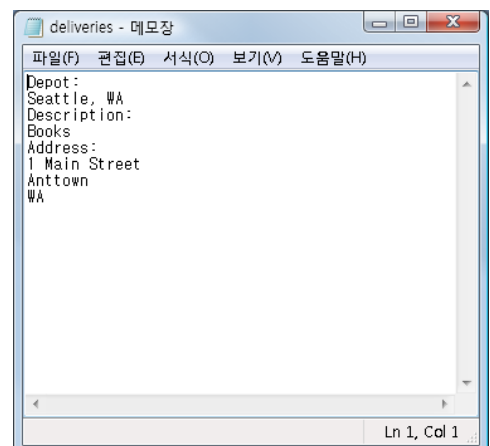
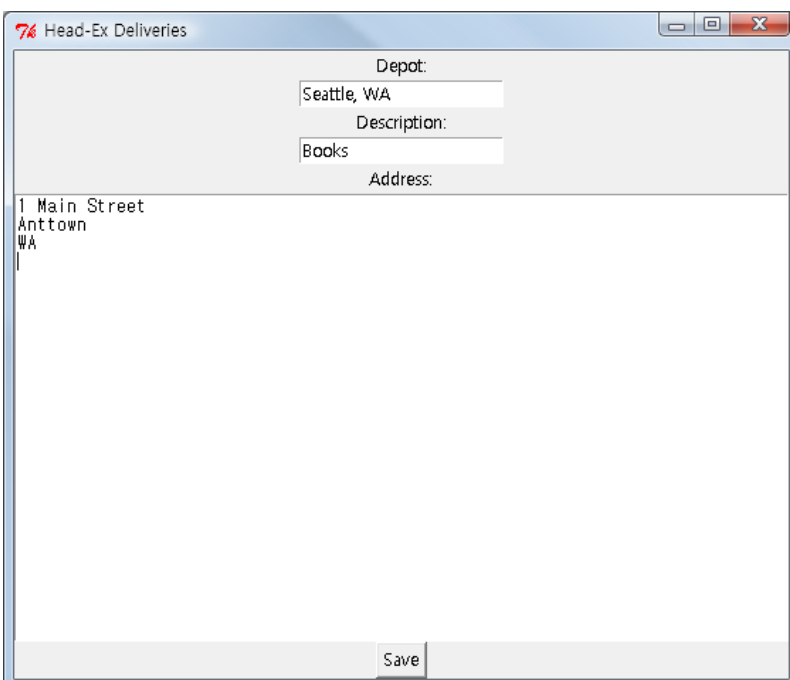
Label(app, text = "Address:").pack()
address = Text(app)
address.pack()

Button(app, text = "Save", command = save_data).pack()

app.mainloop()
```



시험주행



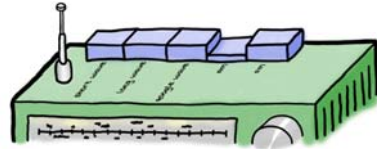
라디오 버튼 위젯 (Radio Button Widget)

❑ 창고(Depot)는 몇 개 안되어서 텍스트로 입력하면 오자로 오류가 발생할 가능성 높음

- 입력 대신에 여러 개 창고 중 하나를 선택하는 코드로 변경

❑ 라디오 버튼 위젯

- 그룹 중 하나의 항목을 선택



- Radiobutton(app, text="Cambridge, MA").pack()
- Radiobutton(app, text="Cambridge, UK").pack()
- Radiobutton(app, text="Seattle, WA").pack()

Cambridge, MA

Cambridge, UK

Seattle, WA

헤데스 배달회사 코드: 라디오 버튼

```
from tkinter import *

def save_data():
    fileD = open("deliveries.txt", "a")
    fileD.write("Depot:\n")
    fileD.write("%s\n" % depot.get())
    fileD.write("Description:\n")
    fileD.write("%s\n" % description.get())
    fileD.write("Address:\n")
    fileD.write("%s\n" % address.get("1.0", END))
    depot.delete(0, END)
    description.delete(0, END)
    address.delete("1.0", END)

app = Tk()
app.title('Head-Ex Deliveries')
```

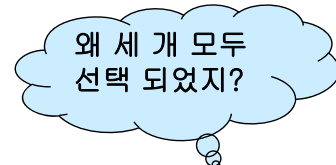
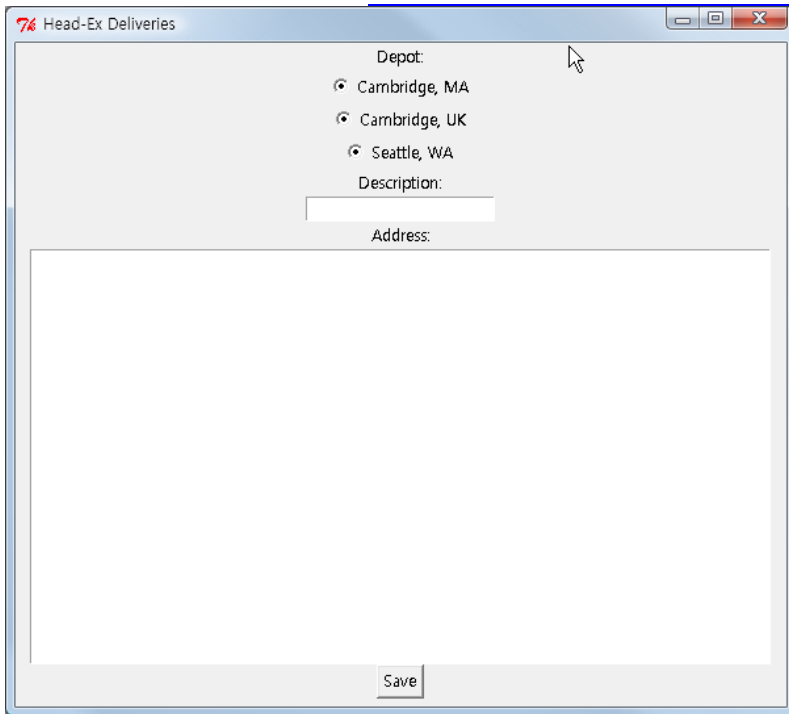
```
Label(app, text = "Depot:").pack()
Radiobutton(app, text="Cambridge, MA").pack()
Radiobutton(app, text="Cambridge, UK").pack()
Radiobutton(app, text="Seattle, WA").pack()
```

```
Label(app, text = "Description:").pack()
description = Entry(app)
description.pack()
```

```
Label(app, text = "Address:").pack()
address = Text(app)
address.pack()
```

```
Button(app, text = "Save", command =
save_data).pack()
```

```
app.mainloop()
```



- ❑ 세 개의 라디오 버튼이 상호 연동되어야 함
 - 위젯 간의 동기화 필요
 - 모델 뷰 컨트롤러(Model View Controller, MVC)
- ❑ 라디오 버튼이 공유하는 제어변수(모델) 생성
 - 텍스트를 저장하는 제어변수(control variable) 생성
 - `service = StringVar()` # 텍스트 제어변수 생성
 - `service.set(None)` # 변수 값이 없음으로 설정
 - `service.get()` # 변수 값 읽기
- ❑ 위젯 생성 시 제어변수와 값을 지정

 First Class

 Next Business Day

```

service = StringVar()
service.set(None)
Radiobutton(app, variable = service, text = "First Class", value = "First Class")
Radiobutton(app, variable = service, text = "Next Business Day", value = "Next Business Day")
print(service.get())
    
```

헤데스 배달회사 코드: 라디오 버튼 수정

```

from tkinter import *

def save_data():
    fileD = open("deliveries.txt", "a")
    fileD.write("Depot:\n")
    fileD.write("%s\n" % depot.get())
    fileD.write("Description:\n")
    fileD.write("%s\n" % description.get())
    fileD.write("Address:\n")
    fileD.write("%s\n" % address.get("1.0",
        END))
    depot.set(None)
    description.delete(0, END)
    address.delete("1.0", END)

app = Tk()
app.title('Head-Ex Deliveries')

Label(app, text = "Depot:").pack()

```

```

depot = StringVar()
depot.set(None)
Radiobutton(app, variable=depot, text="Cambridge, MA",
    value="Cambridge, MA").pack()
Radiobutton(app, variable=depot, text="Cambridge, UK",
    value="Cambridge, UK").pack()
Radiobutton(app, variable=depot, text="Seattle, WA",
    value="Seattle, WA").pack()

Label(app, text = "Description:").pack()
description = Entry(app)
description.pack()

Label(app, text = "Address:").pack()
address = Text(app)
address.pack()

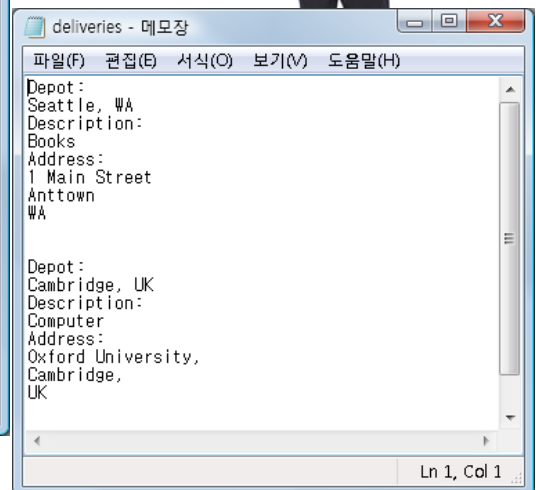
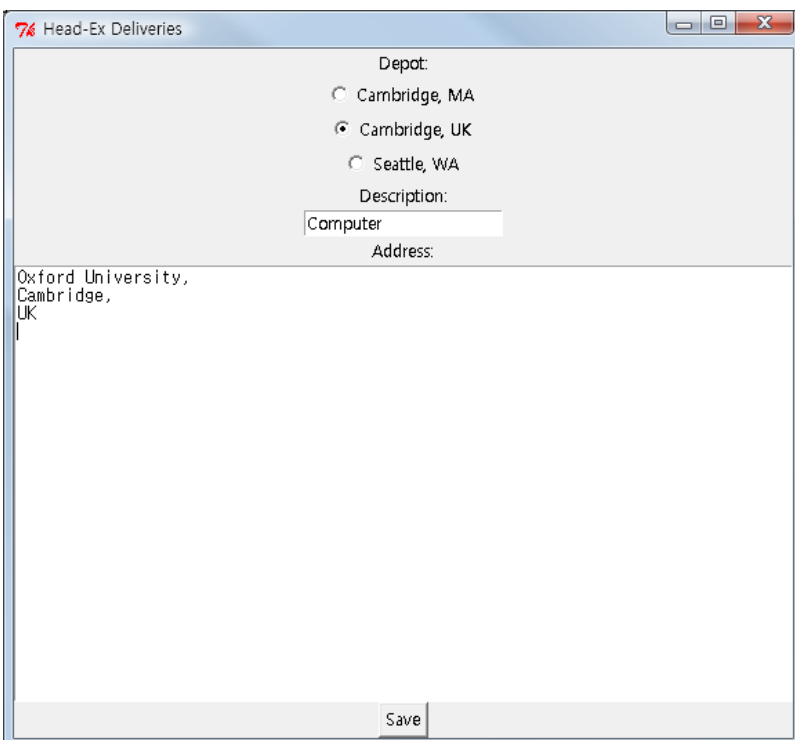
Button(app, text = "Save", command = save_data).pack()

app.mainloop()

```

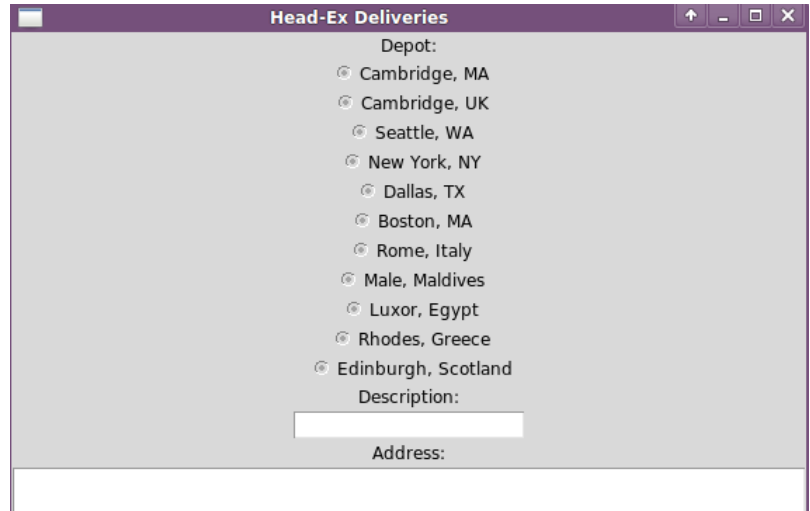


시험주행



너무 많은 창고 선택

- ❑ 헤덱스 사업이 번창하여 세계 각지에 창고를 신설
- ❑ 너무 많은 창고가 만들어져 화면에 라디오 버튼이 너무 많음



옵션 메뉴 위젯 (Option Menu Widget)

- ❑ 옵션 메뉴 위젯(또는 드롭다운 리스트 상자)은 라디오 버튼과 같이 많은 항목 중 하나를 선택하지만 다음과 같이 차이

- 더 적은 공간을 차지
- 클릭하면 펼쳐져서 많은 항목을 보여줌
- 위젯 생성

```
OptionMenu(app, depot, "Cambridge, MA", "Cambridge, UK", "Seattle, WA").pack()
```

- 두 번째 인자가 제어변수
- 세 번째 이 후 인자가 위젯에 보이는 목록들
 - 긴 목록들을 리스트로 참조할 수도 있음



```
depots = ["Cambridge, MA", "Cambridge, UK", "Seattle, WA"]
```

```
depot = StringVar()
```

```
depot.set(None)
```

```
OptionMenu(app, depot, *depots).pack()
```


헤데스 배달회사 코드: 옵션 메뉴 (1)

```

from tkinter import *

def save_data():
    fileD = open("deliveries.txt", "a")
    fileD.write("Depot:\n")
    fileD.write("%s\n" % depot.get())      # 선택된 옵션 목록 읽음
    fileD.write("Description:\n")
    fileD.write("%s\n" % description.get())
    fileD.write("Address:\n")
    fileD.write("%s\n" % address.get("1.0", END))
    depot.set(None)
    description.delete(0, END)
    address.delete("1.0", END)

def read_depots(file): # 옵션 목록을 파일에서 읽어서 리스트로 반환
    depots = []
    depots_f = open(file)
    for line in depots_f:
        depots.append(line.rstrip()) # 줄바꿈 문자를 제거한 후 리스트에 추가
    return depots

```

순천향대

데이터

헤데스 배달회사 코드: 옵션 메뉴 (2)

```

app = Tk()
app.title('Head-Ex Deliveries')

Label(app, text = "Depot:").pack()
depot = StringVar()
depot.set(None)
options = read_depots("depots.txt")
OptionMenu(app, depot, *options).pack()

Label(app, text = "Description:").pack()
description = Entry(app)
description.pack()

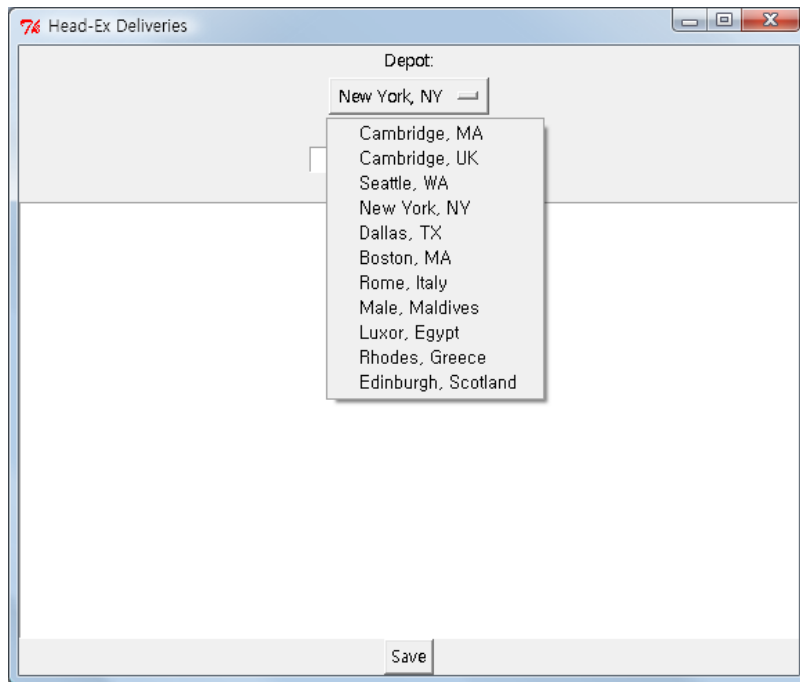
Label(app, text = "Address:").pack()
address = Text(app)
address.pack()

Button(app, text = "Save", command = save_data).pack()
app.mainloop()

```

순천향대

기과 데이터



□ 프로그래밍 도구

- **MVC** – 모델, 뷰, 컨트롤러
 - 모델은 저장된 데이터, 뷰는 위젯, 컨트롤러는 tkinter 코드로 모델이 변경되면 모든 뷰에게 알려줌
- 뷰와 데이터를 분리하는 데이터 모델 사용
- 모델을 공유해야 라디오 버튼이 제대로 작동
- GUI 위젯 항목을 코드에 고정하지 말고 동적으로 지정
 - 파일로 읽어서 옵션 메뉴 목록 지정



프로그래밍 도구상자 (2)

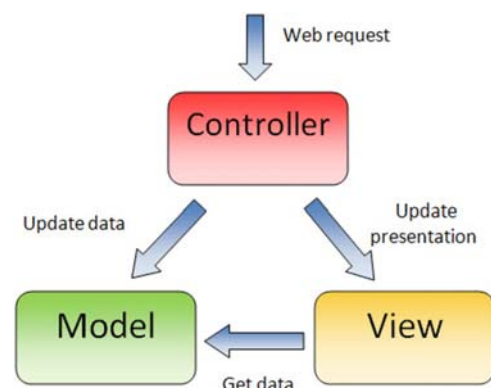
□ 파이썬 도구

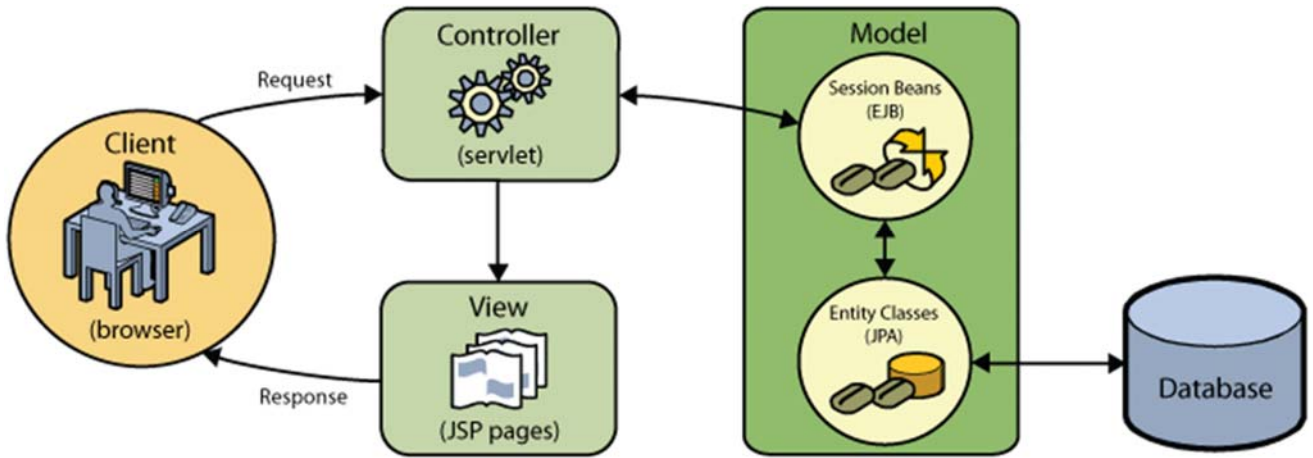
- `Entry()` - tkinter의 입력 위젯, 한라인의 텍스트만 입력
- `Text()` - tkinter의 텍스트 위젯, 한라인의 텍스트만 입력
- 입력 필드는 인덱스가 0부터 시작
- 텍스트 필드는 '행·열' 스트링으로 인덱스하며 "1.0"에서 시작
- tkinter 컨트롤러 - 데이터가 변하면 뷰에게 알려줌
- `StringVar()` - GUI 화면을 제어하고 갱신하는 tkinter 스트링 변수
- `RadioButton()` - tkinter의 라디오 버튼 위젯, 그룹 중 하나의 항목을 선택
- `OptionMenu()` - tkinter의 옵션 메뉴 위젯, 드롭다운 리스트 상자, 많은 항목들 중 하나를 선택할 때 유용함

참고: MVC (Model-View-Controller)

□ MVC는 연산과 처리를 담당하는 프로그램의 논리(logic) 구조에서 사용자에게 제공하는 프리젠테이션(표현부분)을 분리하는 소프트웨어 접근 방법론

- 모델은 애플리케이션의 정보(데이터)의 구조를 표현
- 뷰는 사용자에게 보여질 부분을 표현
 - 일반적으로 웹 페이지
- 컨트롤러는 데이터와 비즈니스 로직 사이의 상호동작을 관리
 - 일반적으로 모델과 뷰사이에서 동작
 - HTTP 요청을 처리하여 웹 페이지를 생성





1. 최종 헤덱스 배달회사 코드를 작성하고 실행
2. 앞에서 배운 내용을 사용한 임의의 프로그램 작성
 - 프로그램 설명
 - 프로그램 소스
 - 실행 결과