# **15차 연습문**항

## **파이썬 심화문항: 그래픽스**

| 수강생 공지 사항   * 제출은 프로그래머스를 통해 해주시기 바랍니다([링크](https://campus.programmers.co.kr/app/courses/24550/curriculum)). * 파일명은 아래와 같은 형식으로 제출해주세요   + 교육생번호\_이름\_교과목\_문항\_N차시.pdf   ex) DR-11111\_홍길동\_파이썬\_연습문항\_1차시.pdf  ex) DR-11111\_홍길동\_파이썬\_추가문항\_1차시.pdf   * 답은 "write your answer"에 적어주세요. 다만 코딩 문제의 경우 output까지 답에 포함시켜 주세요.   ex)     * 답은 캡쳐를 하셔도 되고 텍스트로 넣으셔도 됩니다. * 마감 기한은 문제가 나간 주 **일요일 23:59까지**입니다. |
| --- |

### **1.Numpy는 수치연산을 위한 핵심 라이브러리입니다. Numerical Python의 줄임말로, 다차원 배열을 다루는 기능과 선형 대수 연산과 같은 고수준 수학함수들을 제공합니다. Numpy는 리스트와 비교해 더 빠르고 메모리 효율적인 배열을 제공하며, 대용량 데이터 연산을 보다 효율적으로 수행할 수 있습니다.**

아래의 수식이 제공하는 x, y 랜덤데이타를 가지고 산포그래프를 그려보세요.

x = np.random.normal(size=100) # 정규 분포를 따르는 랜덤 데이터 생성

y = 2.5 \* x + np.random.normal(size=100) # 선형 관계를 가지는 랜덤 데이터 생성

| Example Code |
| --- |
| #no.1  import numpy as np  import matplotlib.pyplot as plt  x=np.random.normal(size=100)  y=2.5\*x+np.random.normal(size=100)  plt.scatter(x,y)  plt.title('Numpy')  plt.xlabel('x')  plt.ylabel('y')  plt.show() |

|  |
| --- |

### **2.Tkinter는 파이썬에서 기본적으로 제공하는 GUI(Graphical User Interface) 라이브러리입니다. Tkinter를 사용해서 'Tkinter GUI를 시작하셨습니다.' 라는 메세지를 보여주는 윈도우를 만들어보세요.**

| Example Code |
| --- |
| #no.2  import tkinter as tk  root = tk.Tk()  root.title('Tkinter')  root.geometry('300x100')  label = tk.Label(root,text='Tkinter GUI를 시작하셨습니다.')  label.pack(pady=20)  root.mainloop() |

|  |
| --- |

### **3.GUI 윈도우에 아래와 같은 조건을 실행하는 프로그램을 작성하시오.**

- GUI 제목은 피트에서 미터로 변환

- 피트 데이터를 입력받은 후, 하단에 위치한 Calculate버튼을 누르면 미터로 변환하여 윈도우상에 보여줌

| Example Code |
| --- |
| #no.3  import sys  from PyQt5.QtCore import \*  from PyQt5.QtGui import \*  from PyQt5.QtWidgets import \*  def calculate():  feet=float(feetInput.text())  meter=feet\*0.3048  result.setText(f'미터: {meter: .4f}')  app = QApplication(sys.argv)  win = QWidget()  win.setWindowTitle('피트에서 미터로 변환')  feetLabel=QLabel('피트')  feetInput=QLineEdit()  bt=QPushButton('Calculate')  bt.clicked.connect(calculate)  result=QLabel('결과: ')  input=QHBoxLayout()  input.addWidget(feetLabel)  input.addWidget(feetInput)  buttonLayout = QHBoxLayout()  buttonLayout.addWidget(bt)  resultLayout = QHBoxLayout()  resultLayout.addWidget(result)  mainLayout = QVBoxLayout()  mainLayout.addLayout(input)  mainLayout.addLayout(buttonLayout)  mainLayout.addLayout(resultLayout)  win.setLayout(mainLayout)  win.show()  sys.exit(app.exec\_()) |

|  |
| --- |

### **4.문제13에서 피트에서 미터로 일방 변환인 프로그램을 피트나 미터 어느 한쪽에 입력이 들어오면 다른 단위로 변환시키는 프로그램으로 변경하고, 그에 맞춰 제목이나 화면 구성을 수정하시오.**

| Example Code |
| --- |
| import sys  from PyQt5.QtCore import \*  from PyQt5.QtGui import \*  from PyQt5.QtWidgets import \*  def calculate():  if feetInput.text() =='':  meter=float(meterInput.text())  feet=meter/0.3048  result.setText(f'피트: {feet: .4f}')  else:  feet=float(feetInput.text())  meter=feet\*0.3048  result.setText(f'미터: {meter: .4f}')  app = QApplication(sys.argv)  win = QWidget()  win.setWindowTitle('피트에서 미터로 변환')  feetLabel=QLabel('피트')  feetInput=QLineEdit()  meterLabel=QLabel('미터')  meterInput=QLineEdit()  bt=QPushButton('Calculate')  bt.clicked.connect(calculate)  result=QLabel('결과: ')  inputfeet=QHBoxLayout()  inputfeet.addWidget(feetLabel)  inputfeet.addWidget(feetInput)  input\_meter=QHBoxLayout()  input\_meter.addWidget(meterLabel)  input\_meter.addWidget(meterInput)  buttonLayout = QHBoxLayout()  buttonLayout.addWidget(bt)  resultLayout = QHBoxLayout()  resultLayout.addWidget(result)  mainLayout = QVBoxLayout()  mainLayout.addLayout(inputfeet)  mainLayout.addLayout(input\_meter)  mainLayout.addLayout(buttonLayout)  mainLayout.addLayout(resultLayout)  win.setLayout(mainLayout)  win.show()  sys.exit(app.exec\_()) |

|  |
| --- |

### **5.문제9의 거북이는 윈도우를 벗어나면 보이지 않는다. 거북이가 윈도우를 벗어나면 반대편으로 나오도록 프로그램을 보완해 보자.**

| Example Code |
| --- |
| # Write your code here |

| write your answer |
| --- |

### **6.Pygame은 파이썬으로 게임을 개발할 때 유용하게 사용되는 크로스 플랫폼 게임 개발 라이브러리입니다. 주로 2D 게임 개발에 많이 사용되며, 다양한 멀티미디어 처리 기능을 제공하여 게임 개발자들이 게임을 만드는 데 필요한 여러 기능을 간편하게 구현할 수 있도록 돕습니다. Pygame을 사용하여 윈도우를 생성하고 배경색 을 설정하는 프로그램을 작성해 보세요.**

- Pygame을 설치하려면 명령 프롬프트나 터미널에서 다음 명령을 실행합니다.

. python -m pip install --upgrade pip

. pip install pygame

| Example Code |
| --- |
| # Write your code here |

| write your answer |
| --- |

### **7.생성된 윈도우안에서 공이 움직이다가 벽을 만나면 튀어 나가는 프로그램을 작성해 봅니다.**

- 공이 화면 중앙에 멈춰 있다가 아무 키나 입력되면 움직이기 시작함.

- 장애물(벽)을 만나면 입사각과 같은 각도로 반사됨

- 게임엔진은 그래픽 처리와 물리 시뮬레이션, 게임 루프등과 같은 일반 소프트웨어하고 다른 구조적 차이가 있음

| Example Code |
| --- |
| # Write your code here |

| write your answer |
| --- |