Reading and Debugging Python Code

- Students will identify the parts of a Python program
- Students will describe the history of the computer bug.
- Students will look at debugging techniques and classifying errors.
- Students will compare different kinds of errors

10/8/18

Journal

Name 2 things from the glossary that are new to you about python. Be ready to share them out.

Feeback From Friday?

Eliza – like program

Sample python code

Let’s mark this up to clarify glossary terms

```
# Eliza-like program
def interview(questions):
   responseNumber = 0    # start with the first response
   userResponse = input( "? " ) # ask the person to type an answer
   while ( not "bye" in userResponse):
      print('\n\n' + questions[ responseNumber ] + " "+ userResponse ,end="")
      userResponse = input( "? " )
      responseNumber+=1
      if responseNumber==len(questions):
         responseNumber = 0

def main():
       print('Welcome, I am concerned about you and have a few questions. You can leave by saying "bye".')
       print("Something seems to be troubling you.  How do you feel",end="")
       responses = [
          "Interesting,Let's dive deeper.\nWhat other feelings come to mind when you think about feeling",
          "Really, how did you feel before you felt",
          "I wonder, do you think a lot about how you used to feel"
          ]
       interview(responses) # call the interview function using the responses defined above
```

```
# Eliza-like program
def interview(questions):
    responseNumber = 0    # start with the first response
    userResponse = input( "? " ) # ask the person to type an answer
    while ( not "bye" in userResponse):
        print('\n\n' + questions[ responseNumber ] + " "+ userResponse ,end="")
        userResponse = input( "? " )
        responseNumber+=1
        if responseNumber==len(questions):
            responseNumber = 0

def main():
        print('Welcome, I am concerned about you and have a few questions. You can leave by saying "bye".')
        print("Something seems to be troubling you.  How do you feel",end="")
        responses = [
            "Interesting,Let's dive deeper.\nWhat other feelings come to mind when you think about feeling",
            "Really, how did you feel before you felt",
            "I wonder, do you think a lot about how you used to feel"
            ]
        interview(responses) # call the interview function using the responses defined above
```

Comments are a note to the humans reading the code.

They can explain what is going on or describe the parts of the program.

```
# Eliza-like program
def interview(questions):
    responseNumber = 0     # start with the first response
    userResponse = input( "? " ) # ask the person to type an answer
    while ( not "bye" in userResponse):
        print('\n\n' + questions[ responseNumber ] + " "+ userResponse ,end="")
        userResponse = input( "? " )
        responseNumber+=1
        if responseNumber==len(questions):
            responseNumber = 0

def main():
        print('Welcome, I am concerned about you and have a few questions. You can leave by saying "bye".')
        print("Something seems to be troubling you.  How do you feel",end="")
        responses = [
            "Interesting,Let's dive deeper.\nWhat other feelings come to mind when you think about feeling",
            "Really, how did you feel before you felt",
            "I wonder, do you think a lot about how you used to feel"
            ]
        interview(responses) # call the interview function using the responses defined above
```

~~Cross out all of the comments.~~

The computer ignores them.

What symbol do these comments start with that sets them apart from the code that the computer needs to understand?

#

```
~~# Eliza-like program~~
def interview(questions):
    responseNumber = 0    ~~# start with the first response~~
    userResponse = input( "? " ) ~~# ask the person to type an answer~~
    while ( not "bye" in userResponse):
        print('\n\n' + questions[ responseNumber ] + " "+ userResponse ,end="")
        userResponse = input( "? " )
        responseNumber+=1
        if responseNumber==len(questions):
            responseNumber = 0

def main():
    print('Welcome, I am concerned about you and have a few questions. You can leave by saying "bye".')
    print("Something seems to be troubling you.  How do you feel",end="")
    responses = [
        "Interesting,Let's dive deeper.\nWhat other feelings come to mind when you think about feeling",
        "Really, how did you feel before you felt",
        "I wonder, do you think a lot about how you used to feel"
        ]
    interview(responses) ~~# call the interview function using the responses defined above~~
```

Functions are parts of a program that has been divided up into sections.

You define a function

def is short for define

How many functions are in this program?

```
~~# Eliza-like program~~
def interview(questions):
    responseNumber = 0     ~~# start with the first response~~
    userResponse = input( "? " ) ~~# ask the person to type an answer~~
    while ( not "bye" in userResponse):
        print('\n\n' + questions[ responseNumber ] + " "+ userResponse ,end="")
        userResponse = input( "? " )
        responseNumber+=1
        if responseNumber==len(questions):
            responseNumber = 0

def main():
        print('Welcome, I am concerned about you and have a few questions. You can leave by saying "bye".')
        print("Something seems to be troubling you.  How do you feel",end="")
        responses = [
            "Interesting,Let's dive deeper.\nWhat other feelings come to mind when you think about feeling",
            "Really, how did you feel before you felt",
            "I wonder, do you think a lot about how you used to feel"
            ]
        interview(responses) ~~# call the interview function using the responses defined above~~
```

Underline the names of the functions

What does each function in this program do?

The code that is part of a function is always indented.

Draw boxes around the code in the functions

```
~~# Eliza-like program~~
def interview(questions):
    responseNumber = 0    ~~# start with the first response~~
    userResponse = input( "? " ) ~~# ask the person to type an answer~~
    while ( not "bye" in userResponse):
        print('\n\n' + questions[ responseNumber ] + " "+ userResponse ,end="")
        userResponse = input( "? " )
        responseNumber+=1
        if responseNumber==len(questions):
            responseNumber = 0

def main():
        print('Welcome, I am concerned about you and have a few questions. You can leave by saying "bye".')
        print("Something seems to be troubling you.  How do you feel",end="")
        responses = [
            "Interesting,Let's dive deeper.\nWhat other feelings come to mind when you think about feeling",
            "Really, how did you feel before you felt",
            "I wonder, do you think a lot about how you used to feel"
            ]
        interview(responses)~~# call the interview function using the responses defined above~~
```

Input statements wait for the human using the program to type in a response

Where are the 2 places in the code where the computer will wait for a person to type in an answer?

```
~~# Eliza-like program~~
def interview(questions):
   responseNumber = 0    ~~# start with the first response~~
   userResponse = input( "? " ) ~~# ask the person to type an answer~~
   while ( not "bye" in userResponse):
      print('\n\n' + questions[ responseNumber ] + " "+ userResponse ,end="")
      userResponse = input( "? " )
      responseNumber+=1
      if responseNumber==len(questions):
         responseNumber = 0

def main():
       print('Welcome, I am concerned about you and have a few questions. You can leave by saying "bye".')
       print("Something seems to be troubling you.  How do you feel",end="")
       responses = [
          "Interesting,Let's dive deeper.\nWhat other feelings come to mind when you think about feeling",
          "Really, how did you feel before you felt",
          "I wonder, do you think a lot about how you used to feel"
          ]
       interview(responses) ~~# call the interview function using the responses defined above~~
```

Circle the input statements

```
~~# Eliza-like program~~
def interview(questions):
    responseNumber = 0    ~~# start with the first response~~
    userResponse = input( "? " ) ~~# ask the person to type an answer~~
    while ( not "bye" in userResponse):
        print('\n\n' + questions[ responseNumber ] + " "+ userResponse ,end="")
        userResponse = input( "? " )
        responseNumber+=1
        if responseNumber==len(questions):
            responseNumber = 0

def main():
        print('Welcome, I am concerned about you and have a few questions. You can leave by saying "bye".')
        print("Something seems to be troubling you.  How do you feel",end="")
        responses = [
            "Interesting,Let's dive deeper.\nWhat other feelings come to mind when you think about feeling",
            "Really, how did you feel before you felt",
            "I wonder, do you think a lot about how you used to feel"
            ]
        interview(responses) ~~# call the interview function using the responses defined above~~
```

Output statements are messages from the computer to the person using the program. In a simple program they are messages that are printed to the screen.

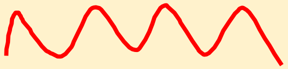
How many print statements are there?

CS Matters

```
~~# Eliza-like program~~
def interview(questions):
    responseNumber = 0    ~~# start with the first response~~
    userResponse = input( "? " ) ~~# ask the person to type an answer~~
    while ( not "bye" in userResponse):
        print('\n\n' + questions[ responseNumber ] + " "+ userResponse ,end="")
        userResponse = input( "? " )
        responseNumber+=1
        if responseNumber==len(questions):
            responseNumber = 0

def main():
    print('Welcome, I am concerned about you and have a few questions. You can leave by saying "bye".')
    print("Something seems to be troubling you.  How do you feel",end="")
    responses = [
        "Interesting,Let's dive deeper.\nWhat other feelings come to mind when you think about feeling",
        "Really, how did you feel before you felt",
        "I wonder, do you think a lot about how you used to feel"
        ]
    interview(responses)~~# call the interview function using the responses defined above~~
```

Draw a zigzag line under each output statement



```
~~# Eliza-like program~~
def interview(questions):
    responseNumber = 0    ~~# start with the first response~~
    userResponse = input( "? " ) ~~# ask the person to type an answer~~
    while ( not "bye" in userResponse):
        print('\n\n' + questions[ responseNumber ] + userResponse ,end="")
        userResponse = input( "? " )
        responseNumber+=1
        if responseNumber==len(questions):
            responseNumber = 0

def main():
    print('Welcome, I am concerned about you and have a few questions. You can leave by saying "bye".')
    print("Something seems to be troubling you.  How do you feel",end="")
    responses = [
        "Interesting,Let's dive deeper.\nWhat other feelings come to mind when you think about feeling",
        "Really, how did you feel before you felt",
        "I wonder, do you think a lot about how you used to feel"
        ]
    interview(responses) ~~# call the interview function using the responses defined above~~
```

camel case

responseNumber = responseNumber + 1

userResponse

A loop is a part of the code that repeats

In this program, there is a loop that keeps repeating as long as the answer to the computer’s question is not “bye”

Draw an arrow to show where the end of the loop goes back up to repeat

What is Eliza?

Matters

Make your own

- "Is Eliza Human?"
    - Use Python to build a chatbot.
    - printing, user input, variables, if statements, while loops