**File: q4.py**

'''

Answer for Question 4 - The Training

Name:

SID:

unikey:

'''

import getpass

import sys

def escape\_key():

while True:

char = getpass.getpass(prompt='', stream=sys.stderr)

if char == '\x1b':

char = getpass.getpass(prompt='', stream=sys.stderr)

if char == '\r':

sys.exit()

break

def intro():

print("Larry: I'm Larry. I'll be your hunting instructor.")

def travel\_to\_camp():

print("Larry: Let's go to the Meadow to begin your training!")

input("Press Enter to travel to the Meadow...")

print("""Travelling to the Meadow...

Larry: This is your camp. Here you'll set up your mouse trap.""")

def setup\_trap() -> tuple:

print("Larry: Let's get your first trap...")

input("Press Enter to view traps that Larry is holding...")

trap = str(input('''Larry is holding...

Left: High Strain Steel Trap

Right: Hot Tub Trap

Select a trap by typing "left" or "right": ''')).strip()

if trap.lower() == "left" or trap.lower() == "right":

if trap.lower() == "left":

trap = "High Strain Steel Trap"

else:

trap = "Hot Tub Trap"

cheese = 1

print(f'''Larry: Excellent choice.

Your "{trap}" is now set!

Larry: You need cheese to attract a mouse.

Larry places one cheddar on the trap!''')

else:

trap = "Cardboard and Hook Trap"

cheese = 0

print("""Invalid command! No trap selected.

Larry: Odds are slim with no trap!""")

return cheese

def sound\_horn() -> str:

print("Sound the horn to call for the mouse...")

horn\_input = input('Sound the horn by typing "yes": ')

return horn\_input

def basic\_hunt(cheese: int, horn\_input: str) -> bool:

if cheese == 0 and horn\_input.lower() != "yes":

print("Nothing happens.")

hunt\_status = False

if cheese == 1 and horn\_input.lower() == "yes":

print("""Caught a Brown mouse!

Congratulations. Ye have completed the training.""")

hunt\_status = True

if cheese == 0 and horn\_input.lower() == "yes":

print("""Nothing happens.

To catch a mouse, you need both trap and cheese!""")

hunt\_status = False

if cheese == 1 and horn\_input.lower() != "yes":

print("""Nothing happens.

To catch a mouse, you need both trap and cheese!""")

hunt\_status = False

return hunt\_status

def end(hunt\_status: bool):

if hunt\_status == True:

print("Good luck~")

def main():

intro()

travel\_to\_camp()

end(basic\_hunt(setup\_trap(), sound\_horn()))

'''

Call your functions here.

Apart from good design, this is so if you import this file in train.py

(question 5), it will not run this code. Because this code's \_\_name\_\_

will not be '\_\_main\_\_', but it will instead be 'q4', allowing you to

import this file to use your functions without running unwanted calling code.

'''

'''

This statement is true if you run this script.

This statement is false if this file is to be imported from another script.

'''

if \_\_name\_\_ == '\_\_main\_\_':

main()

**File: train.py**

'''

Answer for Question 5 - The Training Again

Name:

SID:

unikey:

'''

'''

We recommend you import your 'q4' module to complete this question. It will save

trouble in needing to copy and paste code from previous question. However if you

wish not to, you are free to remove the import below.

'''

from q4 import \*

# you can make more functions here if you please

# or any global variables

def main():

intro()

travel\_to\_camp()

while True:

end(basic\_hunt(setup\_trap(), sound\_horn()))

train = input('''

Press Enter to continue training and "no" to stop training: ''')

if train.lower() == "no":

break

if \_\_name\_\_ == '\_\_main\_\_':

main()

**File: q1.py**

'''

We will use the art by Joan Stark as the logo:

\_\_\_\_()()

/ @@

jgs `~~~~~\\_;m\_\_m.\_>o

Answer for Question 1 - Game Title

Name:

SID:

unikey:

'''

title = "Mousehunt"

logo = '''

\_\_\_\_()()

/ @@

`~~~~~\\_;m\_\_m.\_>o'''

author = 'An INFO1110/COMP9001 Student'

credits = f'''

Inspired by Mousehunt© Hitgrab

Programmer - {author}

Mice art - Joan Stark'''

print(title)

print(logo)

print(credits)

**File: name.py**

'''

Answer for Question 3 - Function

Name:

SID:

unikey:

'''

def is\_valid\_length(name: str) -> bool:

length\_of\_name = len(name)

if name and len(name.strip(' ')) > 0 and length\_of\_name < 10:

is\_valid\_length = True

else:

is\_valid\_length = False

return is\_valid\_length

def is\_valid\_start(name: str) -> bool:

length\_of\_name = len(name)

if name and len(name.strip(' ')) > 0 and length\_of\_name > 0 and name[:1].isalpha():

is\_valid\_start = True

else:

is\_valid\_start = False

return is\_valid\_start

def is\_one\_word(name: str) -> bool:

length\_of\_name = len(name)

if name and length\_of\_name > 0 and name.find(" ") == -1:

is\_one\_word = True

else:

is\_one\_word = False

return is\_one\_word

def is\_valid\_name(name: str) -> bool:

if is\_valid\_length(name) == True and is\_valid\_start(name) == True and is\_one\_word(name) == True:

is\_valid\_name = True

else:

is\_valid\_name = False

return is\_valid\_name

**File: game.py**

'''

Answer for Question 7 - PIAT: The Hunt

Name:

SID:

unikey:

'''

'''

Keep this line!

'''

import random

'''

We recommend you import your 'name', 'train' and 'shop' modules to complete this

question. It will save trouble in needing to copy and paste code from previous

questions. However if you wish not to, you are free to remove the imports below.

Feel free to import other modules that you have written.

'''

import name

import train

import shop

import q1

def hunt(gold, cheddar\_amount, points):

'''

Handles the hunt mechanic.

It includes the inputs and outputs of sounding the horn, the result of

the hunt, the gold and points earned, and whether users want to continue

after failing consecutively.

Parameters:

gold: int, the current quantity of gold the player possesses.

cheddar\_amount: int, the current quantity of cheddar the player possesses.

points: int, the current quantity of points that the player posseses.

Returns:

gold: int, the updated quantity of gold after the hunt.

cheddar\_amount: int, the updated quantity of cheddar after the hunt.

points: int, the updated quantity of points after the hunt.

'''

fails = 0

while True:

if fails == 5:

stop = input('Do you want to continue to hunt? ')

# exit condition

if stop == 'no':

return gold, cheddar\_amount, points

fails = 0

print('Sound the horn to call for the mouse...')

horn = input('Sound the horn by typing "yes": ')

# exit condition

if horn == 'stop hunt':

return gold, cheddar\_amount, points

if horn != 'yes':

print('Do nothing.')

fails += 1

elif cheddar\_amount <= 0:

print('Nothing happens. You are out of cheese!')

fails += 1

else:

cheddar\_amount -= 1

p = random.random()

if p > 0.5:

print('Nothing happens.')

fails += 1

else:

print('Caught a Brown mouse!')

fails = 0

gold += 125

points += 115

print(f'My gold: {gold}, My points: {points}')

print()

def main():

'''

Implement your code here.

'''

# Initialise our amounts for each variable

gold = 125

cheddar\_amount = 0

points = 0

# Use the game title from Q1

q1.main()

print()

hunter\_name = input("What's ye name, Hunter?\n")

# default name to Bob if not valid

if not name.is\_valid\_name(hunter\_name):

hunter\_name = 'Bob'

print(f'Welcome to the Kingdom, Hunter {hunter\_name}!')

print("Before we begin, let's train you up!")

train\_command = input('Press "Enter" to start training or "skip" to Start Game: ')

if train\_command == 'skip':

trap\_name = "Cardboard and Hook Trap"

else:

# we need to print an extra newline if we don't skip training

print()

trap\_name = train.train()

print()

while True:

print(f'What do ye want to do now, Hunter {hunter\_name}?')

print('1. Exit game')

print('2. Join the Hunt')

print('3. The Cheese Shop')

option = input()

if option == '1':

return

elif option == '2':

gold, cheddar\_amount, points = hunt(gold, cheddar\_amount, points)

elif option == '3':

# call our shop function which can be used with any gold, cheddar, trap

# we get the return value as we need to know the updated gold and cheddar

gold, cheddar\_amount = shop.enter\_shop(gold, cheddar\_amount, trap\_name)

print()

if \_\_name\_\_ == '\_\_main\_\_':

main()

**File: shop.py**

'''

Answer for Question 6 - PIAT: The Cheese Shop

Name:

SID:

unikey:

'''

def buy\_cheese(gold: int)-> tuple:

'''

Feature for players to buy cheddar from shop

Parameters:

gold: int, amount of gold player has

Returns:

gold\_spent: int, amount of gold spent

cheese\_bought: int, amount of cheese bought

'''

gold\_spent = 0

cheese\_bought = 0

print(f'You have {gold} gold to spend.')

command = input('State [cheese quantity]: ').split(' ')

if len(command) == 1:

if command[0] != 'back':

print('Sorry, did not understand.')

return gold\_spent, cheese\_bought

cheese\_type = command[0]

cheese\_quantity = int(command[1])

if cheese\_type != 'cheddar':

print('Sorry, did not understand.')

return gold\_spent, cheese\_bought

if cheese\_quantity <= 0:

print('Must purchase a positive amount of cheese.')

return gold\_spent, cheese\_bought

# calculate price

price = cheese\_quantity \* 10

# ensure we are not over spending (buying more than we can)

# e.g. gold is 100, and gold\_spent is currently 70

# if price is 40, then we should not be able to buy it

if gold\_spent + price <= gold:

print(f'Successfully purchase {cheese\_quantity} cheddar.')

# keep track of gold remaining

gold\_spent += price

cheese\_bought += cheese\_quantity

else:

print('Insufficient gold.')

return gold\_spent, cheese\_bought

def display\_inventory(gold: int, cheddar: int, trap: str) -> None:

'''

Prints contents of inventory

Parameters:

gold: int, current gold that player possess

cheddar: int, current amount of cheddar that player possesses

trap: str, current name of trap that player posseses

'''

print(f'Gold - {gold}')

print(f'Cheddar - {cheddar}')

print(f'Trap - {trap}')

def enter\_shop(gold, cheddar, trap):

'''

Runs the shop

This allows modular use of using the shop, as it can be called with any

combination of gold, cheddar and trap from other files (such as game)

Parameters:

gold: int, current gold that player possesses

cheddar: int, current amount of cheddar that player possesses

trap: str, name of trap that player posseses

Returns:

gold: int, updated gold that player possess

cheddar: int, updated amount of cheddar that player possesses

'''

print('Welcome to The Cheese Shop!')

print('Cheddar - 10 gold')

while True:

print()

print('How can I help ye?')

print('1. Buy cheese')

print('2. View inventory')

print('3. Leave shop')

option = input()

if option == '1':

# the function returns a tuple of gold spent and cheddar bought

result = buy\_cheese(gold)

gold -= result[0]

cheddar += result[1]

elif option == '2':

display\_inventory(gold, cheddar, trap)

elif option == '3':

break

return gold, cheddar

def main():

'''

Implement your code here.

'''

# we set our initial values here

gold = 125

cheddar = 0

trap = 'Cardboard and Hook Trap'

# notice we do not get the return value from the enter\_shop function

# this is because we exit the program once we leave the shop

# so we wouldn't need to know the gold and cheddar after exiting the shop

enter\_shop(gold, cheddar, trap)

if \_\_name\_\_ == '\_\_main\_\_':

main()

**File: fe.py**

'''

Write your solution for Question 1 and Question 2 here.

You may add additional functions but the existing functions given in the scaffold must exists.

Name:

SID:

unikey:

'''

def refresh\_file(filename):

'''

Creates an empty file if filename exists.

Parameters:

filename: str, abs path to file.

'''

pass

def log\_events(event, filename):

'''

Write the received event into filename.

If filename argument is missing, default file: /home/saved/temp.txt

Parameters:

event: str, event to be written to file

filename: str, abs path to file.

Returns:

success: bool, True if written successfully. Else False.

'''

pass

def analyze\_game(fobj):

'''

Analyzes the contents of an open file object to extract game information.

The parameter is NOT a str object!

Parameters:

fobj: open file object in read mode

Returns:

output: str, formatted string displaying the game analysis results.

'''

pass

def main(args):

'''

Checks if game analysis feature works correctly

Parameters:

args: list of command line arguments

Returns:

result: str, formatted string displaying the game analysis results

'''

pass

if \_\_name\_\_ == "\_\_main\_\_":

pass

**File: bob.txt**

Start game

Start shop

Bought 10 cheddar

End shop

Start hunt

Nothing happens.

Do nothing

Do nothing

Caught a Brown mouse!

Nothing happens.

Nothing happens.

Nothing happens.

Nothing happens.

Nothing happens.

Caught a Brown mouse!

Nothing happens.

Nothing happens.

Nothing happens. You are out of cheese!

Nothing happens. You are out of cheese!

End hunt

End game

**File: multiple\_fences.txt**

Start game

End game

Start game

Start shop

Bought 10 cheddar

End shop

Start hunt

Nothing happens.

Do nothing

Do nothing

Caught a Brown mouse!

Nothing happens.

Nothing happens.

Nothing happens.

Nothing happens.

Nothing happens.

Caught a Brown mouse!

Nothing happens.

Nothing happens.

Nothing happens. You are out of cheese!

Nothing happens. You are out of cheese!

End hunt

End game

**File: COMP9001.txt**

Start game

Start shop

Bought 1 cheddar

Bought 2 cheddar

Bought 3 cheddar

Bought 4 cheddar

Bought 0 cheddar

End shop

Start hunt

Caught a Brown mouse!

Caught a Brown mouse!

Caught a Brown mouse!

Caught a Brown mouse!

Nothing happens.

Caught a Brown mouse!

Nothing happens.

Caught a Brown mouse!

Caught a Brown mouse!

Nothing happens.

Nothing happens. You are out of cheese!

Nothing happens. You are out of cheese!

End hunt

Start shop

Bought 9 cheddar

Bought 10 cheddar

Bought 11 cheddar

Bought 12 cheddar

Bought 13 cheddar

End shop

Start hunt

Caught a Brown mouse!

Nothing happens.

Nothing happens.

Caught a Brown mouse!

Caught a Brown mouse!

Nothing happens.

Caught a Brown mouse!

Caught a Brown mouse!

Nothing happens.

Nothing happens.

Nothing happens.

Caught a Brown mouse!

Nothing happens.

Nothing happens.

Nothing happens.

Nothing happens.

Caught a Brown mouse!

Nothing happens.

Nothing happens.

Do nothing

Nothing happens.

Nothing happens.

Caught a Brown mouse!

Caught a Brown mouse!

Caught a Brown mouse!

Caught a Brown mouse!

Caught a Brown mouse!

Nothing happens.

Nothing happens.

Caught a Brown mouse!

Caught a Brown mouse!

End hunt

Start shop

End shop

End game

**File: Outside.txt**

Start game

Start shop

Bought 10 cheddar

End shop

Bought 100 cheddar

Start hunt

Nothing happens.

Do nothing

Do nothing

Caught a Brown mouse!

Nothing happens.

Nothing happens.

Nothing happens.

Nothing happens.

Nothing happens.

Caught a Brown mouse!

Nothing happens.

Nothing happens.

Nothing happens. You are out of cheese!

Nothing happens. You are out of cheese!

End hunt

Do nothing

End game

Caught a Brown mouse!