Q y :
Module1: a =
float(input("Enter a temperature in degrees Fahrenheit: "))
print("The temperature in degrees Celsius is " + str((a-32)*(5/9)))

Q y :
Module2: a = input("Enter a
word: ")
a_lower = a.lower()
b = a_lower.count("z")
if b >= 1:
print("Your word has at least one z.")
if b >= 2:
print("Your word has even more than one z!")

Q y :
Module2: a = input("Enter a
word: ")
if a.isalpha():
a_lower = a.lower()
b = a_lower.count("e")
print("Your word contains the letter e",b,"times.")
if a.isnumeric():
print(a,"is not a word.")

Q y :
Module3: for word in
open("pride_and_prejudice.txt"):
if word.rstrip("\n").startswith("e"):
length = len(word.rstrip("\n"))
print(length)

Q y :
Module4: num_above20 = 0

for value in open("rainfall_Jan2017.txt"):
value_float = float(value)
if value_float > 20:
num_above20 += 1

if num_above20 > 0:
print("There was at least one day with 20mm or more rainfall.")
else:
print("There was no day with 20mm or more rainfall.")

Q y :
Module4: max_sofar = 0
for value in open("min_temperatures_2016.txt"):
value_float = float(value)
max_sofar = max(max_sofar, value_float)

print(max_sofar)

Q y :
Module5: for line in

```
open("jane_eyre.txt"):
line_strip = line.rstrip()
count = 0
for word in line_strip.split():
if not "e" in word:
count += len(word)
print(count)
```

Q y :

```
Module5:                                                    for line in
open("jane_eyre_sentences.txt"):
line_strip = line.strip()
words = line_strip.split()
if words[0] == "I" and words[-1] == "me":
print(words[1], words[-2])
```

Q y :

```
Module6:                                                    for line in
open("jane_eyre_sentences.txt"):
line_strip = line.rstrip()
words = line_strip.split()
if len(words) > 10:
print(" ".join(words[:3]) + " [..] " + " ".join(words[-3:]))
```

Q y :

```
Module6:                                                    count = 0
for line in open("jane_eyre_sentences.txt"):
if ";" in line:
line_split = line.split(";")
words_before = line_split[0].split()
words_after = line_split[1].split()
print("Line " + str(count) + ": "+str(len(words_before)) + \
";" + str(len(words_after)))
count += 1
```

Q y :

```
Module7:                                                    is_first_line = True

for row in open("climate_data_Dec2017.csv"):
if is_first_line:
is_first_line = False
else:
values = row.split(",")
wind_speed = float(values[10])
if wind_speed > 65:
print(values[0], values[2], wind_speed)
```

Q y :

```
Module8:                                                    monthly_rainfall = {}
is_first_line = True

for row in open("climate_data_2017.csv"):
if is_first_line:
is_first_line = False
else:
```

```
values = row.split(",")
date = values[0]
month = date.split("-")[1]
state = values[1]
rainfall = float(values[6])

if (month,state) in monthly_rainfall:
monthly_rainfall[(month,state)].append(rainfall)
else:
monthly_rainfall[(month,state)] = [rainfall]

# Build a dictionary of minimum rainfalls for each month
min_monthly_rainfall = {}
for key in monthly_rainfall:
rainfalls = monthly_rainfall[key]
min_monthly_rainfall[key] = min(rainfalls)

# Aggregate the minimum monthly rainfalls to find the highest value
highest_min_monthly_rainfall = -1
highest_month = ""
highest_state = ""
for key in min_monthly_rainfall:
min_rainfall = min_monthly_rainfall[key]
if min_rainfall > highest_min_monthly_rainfall:
highest_min_monthly_rainfall = min_rainfall
highest_month = key[0]
highest_state = key[1]

print("Month:", highest_month)
print("State:", highest_state)
```

Q y :

```
x = {}
first_line = True

for row in open("climate_data_2017_numeric.csv"):
if first_line:
first_line = False
field_names = row.strip().split(",")
else:
value = row.strip().split(",")
city = value[1]
date = value[0]
month = date.split("-")[1]

if month == "08":
for word in range(1,8):
name = field_names[word]
values = float(value[word])
if name not in x:
x[name] = [values]
else:
x[name].append(values)

print("Available field names:")
```

```
for y in field_names[1:]:
print(y)
field = input("Please enter a field name: ")

if field not in field_names[1:]:
print("Invalid field name entered.")
else:
print("Statistics for field '"+ field+"':")
z = x[field]
print("  Min:",min(z),"Max:",max(z))
```

Q y :

Module9: import pandas as pd

```
df = pd.read_csv('climate_data_Dec2017.csv')
grouped_by_state = df.groupby('State')
max_humidity_by_state = grouped_by_state['9am relative humidity (%)'].max()
for k, v in max_humidity_by_state.to_dict().items():
print(k, ':', float(v))
```