1. Assign the value 7 to the variable guess\_me. Then, write the conditional tests (if, else, and elif) to print the string 'too low' if guess\_me is less than 7, 'too high' if greater than 7, and 'just right' if equal to 7.

**Ans:**

**guess\_me=int(input(“Enter value of guess\_me>”))**

**if guess\_me<7:**

**print("too low")**

**elif guess\_me>7:**

**print("too high")**

**else:**

**print("just right")**

1. Assign the value 7 to the variable guess\_me and the value 1 to the variable start. Write a while loop that compares start with guess\_me. Print too low if start is less than guess me. If start equals guess\_me, print 'found it!' and exit the loop. If start is greater than guess\_me, print 'oops' and exit the loop. Increment start at the end of the loop.

**Ans:**

**guess\_me=int(input('Enter guess\_me value>'))**

**start=int(input('Enter start value>'))**

**while True:**

**if guess\_me>start:**

**print("too low")**

**elif guess\_me==start:**

**print("found it")**

**break**

**else:**

**print('oops!')**

**break**

**start+=1**

1. Print the following values of the list [3, 2, 1, 0] using a for loop.

**Ans**

**l=[3,2,1,0]**

**for i in l:**

**print(i)**

1. Use a list comprehension to make a list of the even numbers in range(10)

**Ans:**

**l = [x for x in range(10) if x%2==0]**

1. Use a dictionary comprehension to create the dictionary squares. Use range(10) to return the keys, and use the square of each key as its value.

**Ans:**

**d1=range(1,10)**

**d2=[i\*i for i in d1]**

**#dictionary = dict(zip(d1,d2))**

**dictionary = {k:v for (k,v) in zip(d1,d2)}**

6. Construct the set odd from the odd numbers in the range using a set comprehension (10).

**Ans:**

**odd\_set ={i for i in range(10) if i%2!=0}**

7. Use a generator comprehension to return the string 'Got ' and a number for the numbers in range(10). Iterate through this by using a for loop.

**Ans:**

**gen = ('Got '+str(i) for i in range(10))**

8. Define a function called good that returns the list ['Harry', 'Ron', 'Hermione'].

**Ans:**

**def good():**

**return ['Harry', 'Ron', 'Hermione']**

**print(good())**

9. Define a generator function called get\_odds that returns the odd numbers from range(10). Use a for loop to find and print the third value returned.

**Ans:**

**def get\_odds():**

**for i in range(10):**

**if i%2!=0:**

**yield i**

**lst = [j for j in get\_odds()]**

**k=0**

**for j in lst:**

**if (k+1)%3==0:**

**print(lst[k])**

**k+=1**

**The above code returns odd numbers between 1 and 10 and return each 3rd element of list containing only odd numbers.**

10. Define an exception called OopsException. Raise this exception to see what happens. Then write the code to catch this exception and print 'Caught an oops'.

**Ans:**

**class OopsException(Exception):**

**pass**

**num=int(input("Enter an even number"))**

**try:**

**if num%2==1:**

**raise OopsException**

**else:**

**print("Number is accepted!")**

**except OopsException:**

**print("Number entered is an odd number")**

11. Use zip() to make a dictionary called movies that pairs these lists: titles = ['Creature of Habit', 'Crewel Fate'] and plots = ['A nun turns into a monster', 'A haunted yarn shop'].

**Ans:**

**titles = ['Creature of Habit', 'Crewel Fate']**

**plots = ['A nun turns into a monster', 'A haunted yarn shop']**

**d = dict(zip(titles,plots))**

**print(d)**