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.

Answer:

guess\_me = 7

if guess\_me<7:

print(“too low”)

elif guess\_me>7:

print(“too high”)

else:

print(“just right”)

2. 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.

Answer:

|  |
| --- |
| guess\_me=7 |
| Start = 1 |
|  |  |

while True:

if guess\_me<7:

print(“too low”)

elif guess\_me=7:

print(“found it!”)

else:

print(“oops”)

break

start +=1

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

numbers = [3, 2, 1, 0]

for i in numbers:

print(i)

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

Answer:

for i in range(10):

if i%2==0:

print(i)

5. 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.

Answer:

dictionary={}

limit =10

for num in range(limit):

dictionary[ num]=num\*num

print(dictionary)

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

Answer:

limit =10

odd={num for num in range(limit) if num%2==1}

print(odd)

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.

Answer:

limit =10

generator = ('Got ' + str(num) for num in range(limit))

for i in generator:

print(i)

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

Answer:

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.

Answer:

limit =10

|  |  |
| --- | --- |
|  | |
|  | get\_odds = (num for num in range(limit) if not num % 2 == 0) | |
|  | | count = 0 |
|  | | for num in get\_odds: |
|  | | if count == 2: |
|  | | print(num) |
|  | | break |
|  | | count += 1 |

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'.

Answer:

Class OopsException(Exception):

pass

def with\_exception(a):

if a<0:

raise OopsException(a)

try:

with\_exception(a)

except OopsExcetion as err:

print(“Caught an oops”)

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'].

Answer:

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

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

movies{}

for title, plot in zip(titles, plots):

movies[title]=plot

print(movies)