|  |  |
| --- | --- |
| **Python if**  Syntax: if test\_condition  statements(s)  # program of if  score = int(input("enter your score : "))  if score >= 50:  print("You have passed the exam")  print("Congratulations")  if score < 50:  print("Sorry, You have failed your exam") | OUTPUT  enter your score : 55  You have passed the exam  Congratulations  enter your score : 26  Sorry, You have failed your exam |
| Syntax: if test\_condition  statements(s)  else:  statements(s)  # program of if\_else  score = 35  if score >= 50:  print("You have passed the exam")  print("Congratulations")  else:  print("Sorry, You have failed your exam") | OUTPUT:  Sorry, You have failed your exam |
| Syntax: if test\_condition1  Statements1  elif test\_condition2  statement2  else:  statement3  # program of if\_elif\_else  score = 105  if score >= 100 or score < 0:  print(“Score is invalid”)  elif score >= 50:  print("You have passed the exam")  print("Congratulations")  else:  print("Sorry, You have failed your exam") | Score is invalid |
| **Python While**  Syntax : while test\_condition:  statement(s)  #program of while loop  count = 0  while count < 5:  #print("I am inside a loop")  #print("Looping is interesting")  Print(count)  count = count + 1 | I am inside a loop  Looping is interesting  I am inside a loop  Looping is interesting  I am inside a loop  Looping is interesting  I am inside a loop  Looping is interesting  I am inside a loop  Looping is interesting |
| number = int(input("Enter a number: "))  count = 1  while count <= 10:  product = number \* count  #print (product)  print(number, "x", count , "=", product)  count = count + 1 | Enter a number: 5  5 x 1 = 5  5 x 2 = 10  5 x 3 = 15  5 x 4 = 20  5 x 5 = 25  5 x 6 = 30  5 x 7 = 35  5 x 8 = 40  5 x 9 = 45  5 x 10 = 50 |
| Do your self  Can you modify the multiplication table program so that we get a multiplication table from 10 t0 1 instead of 1 to 10??  number = int(input("Enter a number: "))  count = 10  while count >= 1:  product = number \* count  print(number, "x", count, "=", product)  count = count – 1 | Output:  Enter a number: 6  6 x 10 = 60  6 x 9 = 54  6 x 8 = 48  6 x 7 = 42  6 x 6 = 36  6 x 5 = 30  6 x 4 = 24  6 x 3 = 18  6 x 2 = 12  6 x 1 = 6 |
| **Python sequence**:  Syntax: for item in sequence:  statement(s)  #Program for for loop  text = "Python"  for character in text:  print(character) | OUTPUT  P  y  t  h  o  n |
| languages = ["English", "French", "German"]  for language in languages:  print(language) | English  French  German |
| **Python range()**  #Program  for count in range(1, 7):  print(count) | 1  2  3  4  5  6 |
| #Program for multiplication table  number = int(input("Enter an integer: "))  for count in range(1, 11):  product = number \* count  print(number, "\*", i, "=", product) | Enter an integer: 9  9 \* 1 = 9  9 \* 2 = 18  9 \* 3 = 27  9 \* 4 = 36  9 \* 5 = 45  9 \* 6 = 54  9 \* 7 = 63  9 \* 8 = 72  9 \* 9 = 81  9 \* 10 = 90 |
| Do youe self:  Can you create a program to find the sum of numbers from 1 to 100.  total = 0  # looping from 1 to 100  for number in range(1, 101):  total = total + number  print(total) | 5050 |
| **Python break and continue**  #program  for item in range(1, 6):  print(item)  break  but if I run following  for item in range(1, 6):  break  print(item)  nothing will be in output | OUTPUT  1 |
| #break with for  for item in range(1, 6):  if item == 3:  break  print(item)  print("The end") | 1  2  The end |
| #break with while  while True:  number = float(input("Enter a number: "))  if number < 0:  break  print("You entered:", number) | Enter a number: 4  You entered: 4.0  Enter a number: 67  You entered: 67.0  Enter a number: -9 |
| #Using continue with for  for i in range(5):  number = float(input("Enter a number: "))  # check if number if negative  if number < 0:  continue  print(number) | Output  Enter a number: 4  You entered: 4.0  Enter a number: 54  You entered: 54.0  Enter a number: -9  Enter a number: 76  You entered: 76.0  Enter a number: 67  You entered: 67.0 |
| Do yourself  Can you create a program so that all items of the languages list are printed except Swift and C++?  languages = ["Python", "Java", "Swift", "C", "C++"]  #Program  languages = ["Python", "Java", "Swift", "C", "C++"]  for language in languages:  if language == "Swift" or language == "C++":  continue  print(language) | Output:  Python  Java  C |
| **Python pass**  number = 5.5  if number > 0.0:  pass  #number = 5.5  #if number > 0.0:  # implement this later | This code will run without any errors. |

Summary

If-elif-else

* The if statement is used to create programs that can make decisions.
* The if statement executes its body only when the test condition is true.
* The if statement can have an optional else clause.
* The body of else is executed if the test condition is false.
* To run a block of code among more than 2 alternatives, we can use elif clause inside the if.

About While –loop

* Loops are used in programming to repeat a block of code.
* The while loop runs continuously until the test condition evaluates to false.
* If the test condition of the loop is never false, the loop runs infinitely until the memory runs out. This is known as the infinite loop.

About For loop

* Loops are used in programming to repeat a block of code.
* A for loop is used to iterate through a sequence.
* If we know the number of iterations of the loop, it’s easier and better to use the for loop.
* range() is a useful function that creates a sequence of numbers. It’s common to use range() in a for loop to iterate the loop a certain number of times.

About Python break and continue

* The break statement terminates the loop immediately when it is encountered.
* The continue statement skips the code after it for that iteration of the loop.
* The break & continue statements are almost always used inside the if..else statement.

Do your self

|  |
| --- |
| WAP to read two numbers from the keyboard and display the larger one on the screen. |
| WAP to find, a given number is PRIME or NOT. |
| WAP to check whether a number is positive or negative or 0. |