Web IDE – Python3 Environment

Accessing the IDE

1. Go to: <https://repl.it/>
2. Select Python3
3. Sign-up / Create an account
4. Make sure you can remember your account information for the rest of the course.

Using the IDE

* Use the black area like a calculator to try simple statements or commands
* Use the white area to create programs with multiple statements

Level 0: Basic Math & Strings

Accessing the Tutorial

* Go to: <http://www.letslearnpython.com/learn/>
* Skip directly to “Lesson 3: Math”

Questions

1. Complete “Lesson 3: Math – Math Basics” by typing the sample commands in the black area of the IDE.
   1. Create your own expression using 5 “+” and “-“ operators.
   2. List your expression and the result below.
2. **1+1**
3. => 2 **1+2**
4. => 3 **1+3**
5. => 4 **1+4**
6. => 5 **1+5**
7. => 6
8. **10-1**
9. => 9 **10-2**
10. => 8 **10-3**
11. => 7 **10-4**
12. => 6 **10-5**
13. => 5
14. Complete “Lesson 3: Math – More Operators” by typing the sample commands in the black area of the IDE.
    1. Create your own expression using 5 “\*” and “/” operators.
    2. List your expression and the result below.
15. **10\*2**
16. => 20 **10\*3**
17. => 30 **10\*4**
18. => 40 **10\*5**
19. => 50 **10\*6**
20. => 60

1. **10/10**
2. => 1.0 **20/10**
3. => 2.0 **30/10**
4. => 3.0 **40/10**
5. => 4.0 **50/10**
6. => 5.0
7. Complete “Lesson 3: Math – More Division” by typing the sample commands in the black area of the IDE.
   1. Create one division expression that gives a whole number answer
   2. And one division expression that gives a decimal number answer.
   3. List your expressions and the results below.
8. **100/10**
9. => 10.0 **100/30**
10. => 3.3333333333333335
11. Complete “Lesson 3: Math – Floats” by typing the sample commands in the black area of the IDE.
    1. Use the “round()” function for the expressions you created in question #3 above.
    2. List your “round()” expressions and the results they return below.
12. **round(100/30)**
13. => 3
14. Read through “Lesson 3: Math – Comparison Operators”.
    1. Why do you think Equals is “==” instead of “=”?
    2. What does “=” mean?

Double equal sign means equal to, and the single equal sign can be transferred to is.

1. Complete “Lesson 3: Math – Practice” and “Lesson 3: Math – Practice Answers” by typing the sample commands in the black area of the IDE.
   1. Create an expression using 5 different operators that returns a “True” result
   2. And an expression using 5 different operators that returns a “False” result.
   3. List your expressions and the results returned below.
2. **5 < 4 + 3**
3. => True **6 < 3 + 9**
4. => True **4 > 1 + 2**
5. => True **5 > 3 - 2**
6. => True **6 > 4 + 1**
7. => True **5 < 1 + 1**
8. => False **5 < 2 \* 2**
9. => False **100 > 100 \* 2**
10. => False **250 < 4 / 2**
11. => False **100 > 0 + 101**
12. => False
13. Complete “Lesson 4: Strings – Strings” and “Lesson 4: Strings – Examples” by typing the sample commands in the black area of the IDE.
    1. Explain why typing “apple” works and why typing apple without quotes gives an error.
14. **apple**
15. Traceback (most recent call last):
16. File "python", line 1, in <module>
17. NameError: name 'apple' is not defined. Has no string.
    1. Also explain why “2 + 5” does not equal 7.
18. **"2+5"**
19. => '2+5'. Equals ‘2+5’ because there is a string instead of numbers.
20. Complete “Lesson 4: Strings – Operators” by typing the sample commands in the black area of the IDE.
    1. Explain why typing “appl” + “e” works and why typing “apple” - “e” gives an error.
21. **"apple" - "e"**
22. Traceback (most recent call last):
23. File "python", line 1, in <module>
24. TypeError: unsupported operand type(s) for -: 'str' and 'str'

Unsupported operand type for -.

* 1. Also explain why “Hello” \* 10 works but why “Hello” / 10 does work.

1. **hello/10**
2. Traceback (most recent call last):
3. File "python", line 1, in <module>
4. NameError: name 'hello' is not defined

Hello is not defined when using the slash.

1. Complete “Lesson 4: Strings – Indexes” by typing the sample commands in the black area of the IDE.
   1. List the letters in your first name and the index for each letter in your first name.
2. **"b"+ "i" + "r" + "k" + "a" + "r" + "a" + "n"**
3. => 'birkaran'
4. Complete “Lesson 4: Strings – Indexes Examples” by typing the sample commands in the black area of the IDE.
   1. Explain why print(“Hello!”[4]) does not print “l”.
5. **(“Hello!”[4])**
6. Traceback (most recent call last):
7. File "python", line 1
8. (“Hello!”[4])
9. ^
10. SyntaxError: invalid character in identifier
    1. What does (“Hay, Bob!”[4]) print? For a hint try print(“Hay, Bob!”[3]) and print(“Hay, Bob!”[5])
11. Complete “Lesson 4: Strings – Rules” by typing the sample commands in the black area of the IDE.
    1. Explain why print(“Hello!”[7]) gives an error.
12. **(“Hello!”[7])**
13. Traceback (most recent call last):
14. File "python", line 1
15. (“Hello!”[7])
16. ^
17. SyntaxError: invalid character in identifier

Gives an syntaxerror for invalid character in identifier