

Tasks – Variables

Practice using variables: assign values, reuse them, and combine them with what you know (print, literals, operators). At least 10 tasks. Create each file, run it, and check the output.

Run scripts with: `python3 script_name.py`

Task 1 – Assign and print (`var_assign.py`)

- Create `var_assign.py` .
- Assign the integer `42` to a variable named `answer` .
- Print `answer` .

Expected output:

```
42
```

Task 2 – String variable (`var_string.py`)

- Create `var_string.py` .
- Assign your name (as a string) to a variable, e.g. `name = "Alice"` .
- Print the variable.

Expected output (example):

```
Alice
```

Task 3 – Reassign a variable (`var_reassign.py`)

- Create `var_reassign.py` .
- Assign `10` to a variable `x` . Print `x` .
- Reassign `x` to `20` . Print `x` again.

Expected output:

```
10
20
```

Task 4 – Two variables, one print (`var_two.py`)

- Create `var_two.py` .
- Assign `3` to `a` and `5` to `b` .
- In one `print()` call, print both variables (e.g. `print(a, b)`). Use default spacing or `sep` if you like.

Expected output:

```
3 5
```

Task 5 – Variables in an expression (`var_expression.py`)

- Create `var_expression.py` .
- Assign `10` to `x` and `3` to `y` .
- Print the result of `x + y` , then print `x - y` (use the same variables).

Expected output:

```
13
7
```

Task 6 – Variable holds result of calculation (`var_result.py`)

- Create `var_result.py` .
- Assign to a variable the result of `100 // 7` (integer division).
- Assign to another variable the result of `100 % 7` .
- Print both variables (first quotient, then remainder).

Expected output:

```
14
2
```

Task 7 – Reuse variable in calculation (`var_reuse.py`)

- Create `var_reuse.py` .
- Assign `7` to `n` . Print `n` .
- Then assign to `n` the value `n + 1` (or `n * 2`). Print `n` again.

Expected output (example for `n + 1`):

```
7
8
```

Task 8 – Several variables, one expression (`var_combined.py`)

- Create `var_combined.py` .
- Assign `2` to `a` , `3` to `b` , and `4` to `c` .
- Print the value of `a * b + c` (using the variables, not the raw numbers). Then print `a * (b + c)` .

Expected output:

```
10
14
```

Task 9 – String and number variables (`var_mixed.py`)

- Create `var_mixed.py` .
- Assign a string to a variable (e.g. `label = "Count"`) and an integer to another (e.g. `value = 5`).
- Print both in one line (e.g. `label` and `value` with a separator like `": "` so it reads `Count: 5`).

Expected output (example):

```
Count: 5
```

Task 10 – Float variable and operators (`var_float.py`)

- Create `var_float.py` .
- Assign `3.5` to `price` and `2` to `quantity` .
- Print `price * quantity` (the total). Use a variable for the result if you want, or print the expression directly.

Expected output:

```
7.0
```

Task 11 – Copy value to another variable (`var_copy.py`)

- Create `var_copy.py` .
- Assign `100` to `original` . Assign `original` to another variable `copy` (so both hold 100).
- Print `original` and `copy` . Then change only `original` to `200` and print both again.

Expected output:

```
100
100
200
100
```

Task 12 – Variable for output message (`var_message.py`)

- Create `var_message.py` .
- Assign the string `"Done!"` to a variable `msg` . Print `msg` three times in a row on one line using `print(msg, msg, msg, sep=" ")` (or similar).

Expected output:

```
Done! Done! Done!
```

Done

You've practiced: assigning to variables, reassigning, using variables in expressions and in `print()` , mixing strings and numbers, and reusing and copying values.