

Module 2: Variables and Data Types

Prof. Dr. Florian Haselbeck
Memoona Shakeel

Weihenstephan-Triesdorf University of Applied Sciences

Learning outcomes

Understand the concept of variables and different types of data used in programming

Key Topics

- What is a variable?
- int · float · str · bool
- Lists and Dictionaries
- Naming rules
- Type classification
- Ag data examples

What is a variable?

A variable is a named location in RAM that stores a value. You give it a name, assign a value, and use it anywhere in your program.

Name: The label you use to refer to the value.

Value: The actual data stored in RAM at that moment.

Type: What kind of data it is: whole number, decimal, text, true/false...

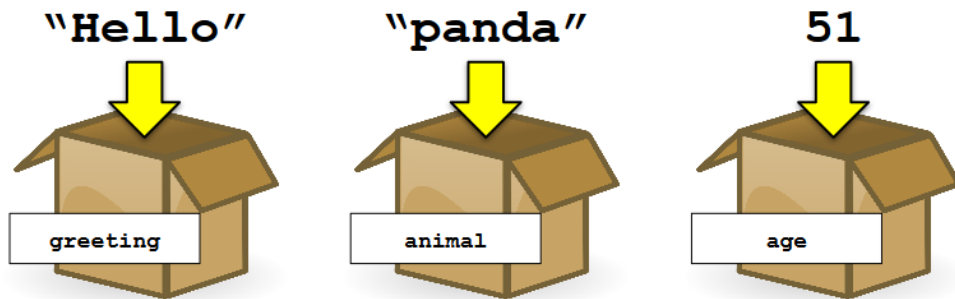
Python detects the type automatically from the value you assign.

assignment

`name = 'alice'`

variable

value



Data Type: int (integer)

What it is: A whole number, no decimal point. Used for counting, IDs, and anything that cannot have a fraction. Operations: + - × ÷ (integer division //) % ** (power). Comparisons: == != < > <= >=

```
# Number of animals on a farm (integers)
cows = 12
chickens = 45
sheep = 20

# Total animals
total_animals = cows + chickens + sheep
print("Total animals:", total_animals) # 77

# Crops harvested (in bags)
corn_bags = 15
wheat_bags = 10

total_bags = corn_bags + wheat_bags
print("Total crop bags:", total_bags) # 25

# Selling animals
cows_sold = 3
cows_left = cows - cows_sold
print("Cows left:", cows_left) # 9
```

Data Type: float (floating-point number)

What it is: A number with a decimal point. Used for measurements, ratios, and anything that needs precision. Always has a decimal point.

Operations: Same as int: + - × /



```
# Milk yield per day
milk_per_cow = 12.5    # float (decimal number)
cows = 4               # int (whole number)

total_milk = milk_per_cow * cows
print("Total milk per day:", total_milk)    # 50.0 (float)
```

Data Type: str (string)

What it is: “Text”, any sequence of characters wrapped in quotes (" " or ' '). Used for names, labels, categories, and messages. It holds Letters, digits, spaces, symbols, and anything in quotes.



```
# Farm name
farm_name = "Green Valley Farm"    # str (text)
# Animal type
animal = "Cow"                     # str
# Milk info
milk_label = "Milk per cow"        # str
# Message
message = "Harvest is ready!"      # str
```

Data Type: bool (boolean)

What it is: A value that is either True or False, nothing else. Used for conditions, flags, and on/off states. It holds exactly two values: **True** or **False**. Used in Logic operations or usually for comparisons.

```

● ● ●
# Is the farm open today?
farm_open = True    # bool

# Did it rain today?
rained_today = False    # bool

# Check if animals need feeding
animals_fed = False    # bool

if animals_fed == False:
    print("Feed the animals!")
```

Collections: list and dict

Key idea: When you need to store multiple values together, use a collection. The two most common in Python are **list** and **dict**.

List (ordered collection): A list stores multiple items in order.

```
# List of animals
animals = ["cow", "chicken", "sheep"] # list of strings

# List of milk yields (liters)
milk_yields = [12.5, 10.0, 9.8] # list of floats
```

Dictionary (key-value pairs): A dictionary (dict) stores data as key-value pairs.

```
# Farm data
farm = {
    "cows": 10, # int
    "milk_per_cow": 12.5, # float
    "farm_name": "Green Farm" # string
}
```

Simple difference: **List** (list) → ordered items → ["cow", "sheep"], **Dict** (dict) → labeled data → {"cows": 10}

Variable naming rules (shorter way to write binary)

Key idea: A good variable name makes your code readable. Python has strict rules, and strong conventions.

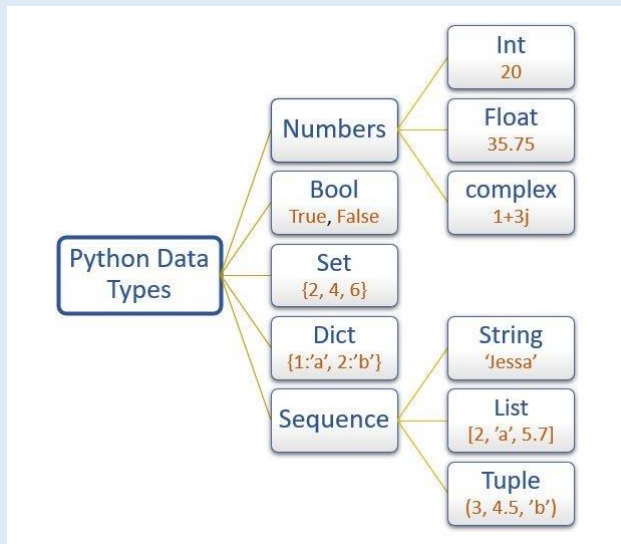
```

# ✅ VALID variable names
farm_name = "Green Farm"
milk_per_cow = 12.5
number_of_cows = 5
total_eggs = 30
is_farm_open = True
crop_yield_kg = 100.5
year2025 = 2025

# ❌ INVALID variable names (examples only – will cause errors if run)
# milk per cow = 12.5      # spaces not allowed
# total-eggs = 30         # hyphens not allowed
# 2cows = 10              # can't start with a number
# farm-name = "A"         # no hyphens
# is farm open = True     # spaces not allowed
# crop yield kg = 100.5   # spaces not allowed
# 2025year = 2025         # can't start with number
```

All data types at a glance

Quick rule: whole number → **int** · decimal → **float** ·
text/name → **str** · yes/no → **bool** · sequence → **list** ·
structured → **dict**



```
# int (integer) → whole numbers
cows = 10
eggs = 25

# float → decimal numbers
milk_per_cow = 12.5
crop_yield = 3.75

# str (string) → text
farm_name = "Green Valley Farm"
animal = "Cow"

# bool (boolean) → True / False
is_raining = False
farm_open = True

# list → multiple items (ordered)
animals = ["cow", "sheep", "goat"]
milk_records = [12.5, 10.0, 9.8]

# dict → key-value pairs
farm = {
    "name": "Green Valley Farm",
    "cows": 10,
    "milk_per_cow": 12.5
}
```

Overview of key content

- A **variable** is a named location in **RAM** that stores a value
- The four basic types: int · float · str · bool
- Lists and dicts let you store multiple values together
- Choosing the right type matters

Thank you!

Contact Detail



Prof. Dr. Florian Haselbeck



florian.haselbeck@hswt.de



Raum D1.230a

For any questions, please feel free to use the [Moodle-Course Forum!](#)