Python Tsunami

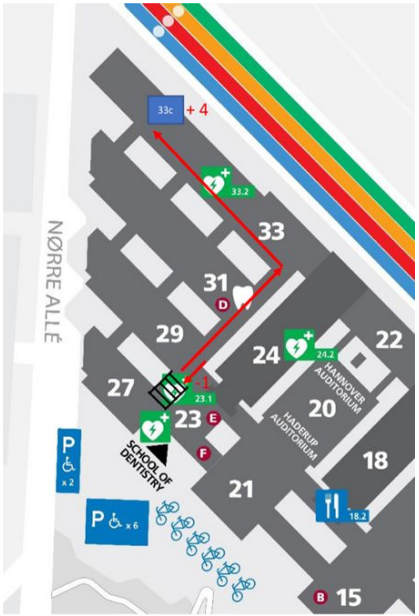
Part 1: Intro to Python by HeaDS

Who are we?

Center for Health Data Science (HeaDS)

The mission of the Center is to strengthen health data science within the Faculty:

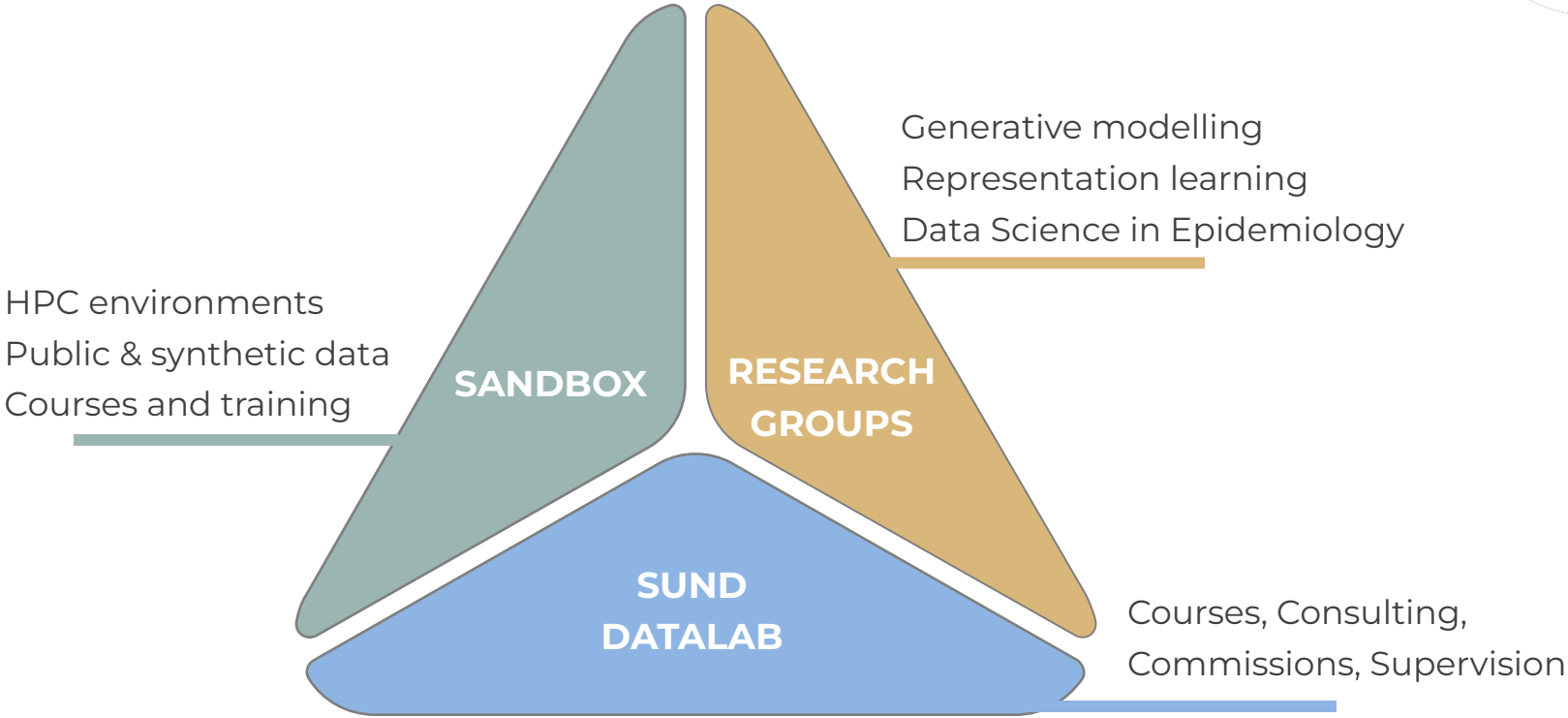
- Active and visible hub for Health Data Science
- Providing data science support for researchers at SUND
- Courses, workshops and training environments to improve data science skills
- Support a network of researchers and educators



https://heads.ku.dk

CENTER STRUCTURE

UCPH HeaDS



DataLab services

UCPH HeaDS

Teaching

Courses

Workshops

Seminars

Consultations

Drop in Thursday

Sparring

Commissioned Projects

Complete Analysis

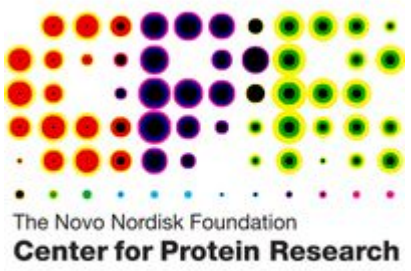
(paid service)

About this course

About this course

Originally developed at the Center for Protein Research (CPR) by:

- Alberto Santos Delgardo (University of Oxford)
- Henry Webel (NNF CPR)
- Annelaura Bach Nielsen (NNF CPR)
- Rita Colaço (PRI)



We say thank you for the course material which we have adapted.

About this course

Your teachers:

Thilde Terkelsen (HeaDS)

Rita Colaço (PRI)

Valentina Sora (HeaDS)

Henrike Zschach (HeaDS)

Inigo Prada Luengo (HeaDS)





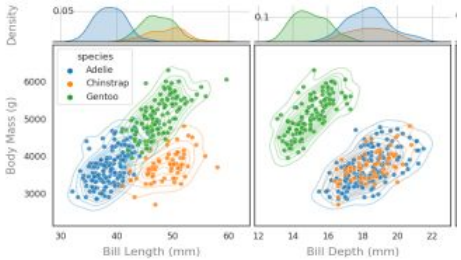
About this course

Starting time	Day 1 (Hannover Aud.)	Day 2 (Holst Aud.)	Day 3 (Holst Aud.)
8:30	Morning coffee (optional)		
8:45	Motivation	Pandas: Series and dataframes, import and examine data, Renaming index/colum	Yesterday Questions + Recap Quiz
9:05	Variables & Data types		
9:45	Coffee break		Coffee break
10	Iterables I: Lists		Visualization
10:15		Coffee break	
10:30		Pandas: Indexing and Selecting Data, Summary functions	
11:30	Iterables II: sets, dicts, tuples		
12:00	Lunch		
13:00	Booleans, operators & conditions	Pandas: Modifying data	Virtual environments and local python installations
13:30			Dataset exercise
14:00		Coffee break	
14:15	Coffee break	Pandas: GroupBy Operations, Sorting and Imputation	
14:30	Loops		
15:30		Pandas Q + A	
16:00	- END -		

What is programming?

What is programming?

Programming is a set of **machine-readable** instructions that transform your input into your desired output.



input → program → output

We will attempt to shed some light on this part

Why is programming nice?

- Learning by doing:

  Difficult to 'break' a computer with wrong programming
- Reproducibility:

  The same thing should happen every time you run (*though some tasks involve some randomness)
- Transferable:

  Easily share your work with colleagues
- Many useful online resources
- Automate complex analysis workflows
- Important tool for working since we live in a data driven world

Why Python?

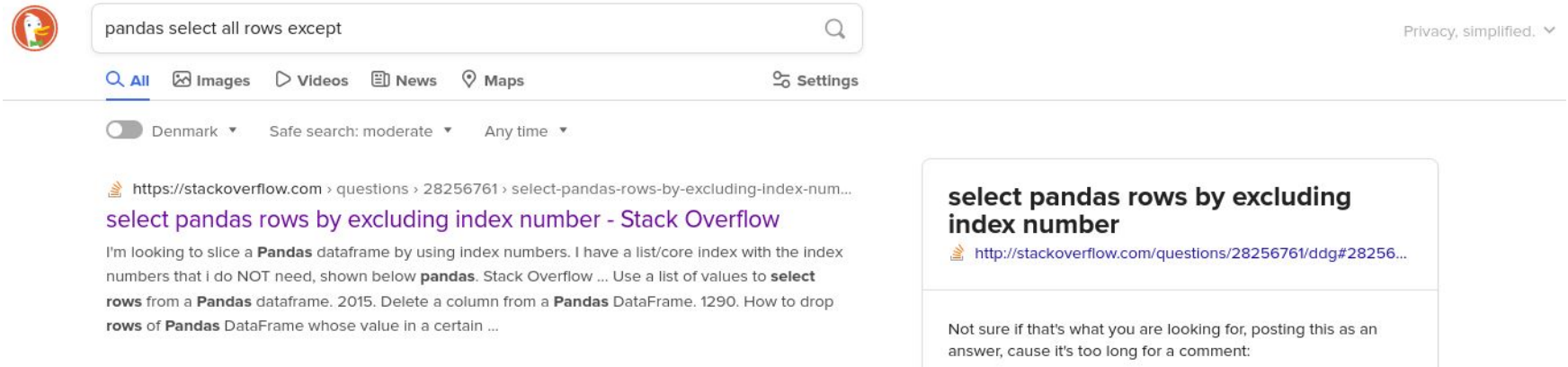
Python is a great programming language for both beginners and advanced programmers:

- Easy to grasp, close to natural language
- Many learning resources available
- Large community (i.e. stackoverflow for questions)
- Libraries
- Can do very advanced things like neural networks

Online communities - Where to get help

Online communities such as stackoverflow are an important tool in programming.

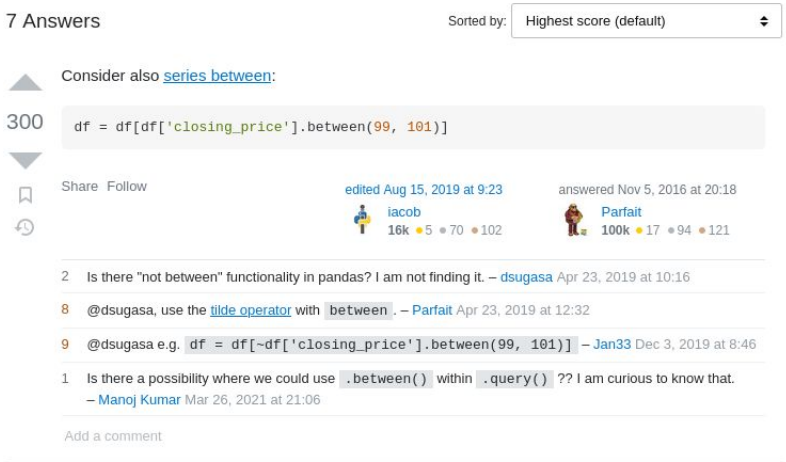
Nobody knows everything, but together we know more than ever before!



Online communities - Where to get help

You have a question ….

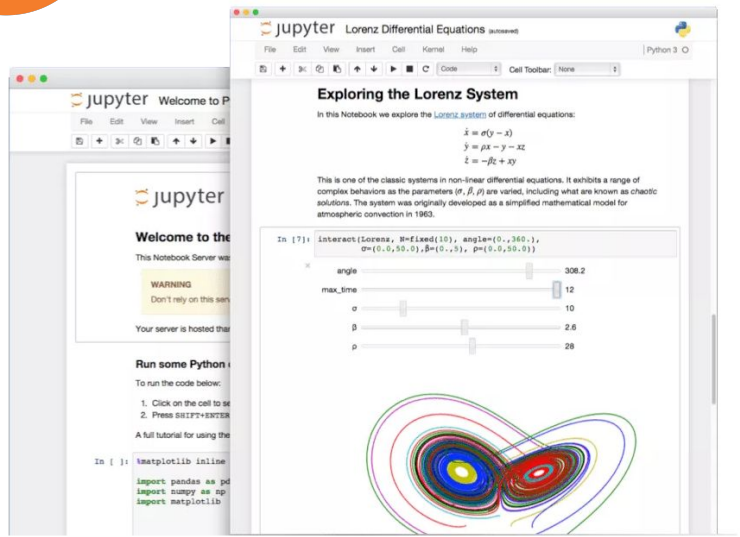
Someone out there has an answer!



Python environments

Jupyter notebook

The Jupyter Notebook is an **open-source application** to create and share documents that contain code, equations, visualizations and text (markdown).



- Browser-based development environment for creating, running and sharing Python code
- Combine code with text and output
- Runs from your **local installation**. I.e., you need Python and the libraries you want to use installed on your computer

Google Colab

Google Colab is a Jupyter Notebook hosted on Google's servers, not your own machine. It still runs in your browser.

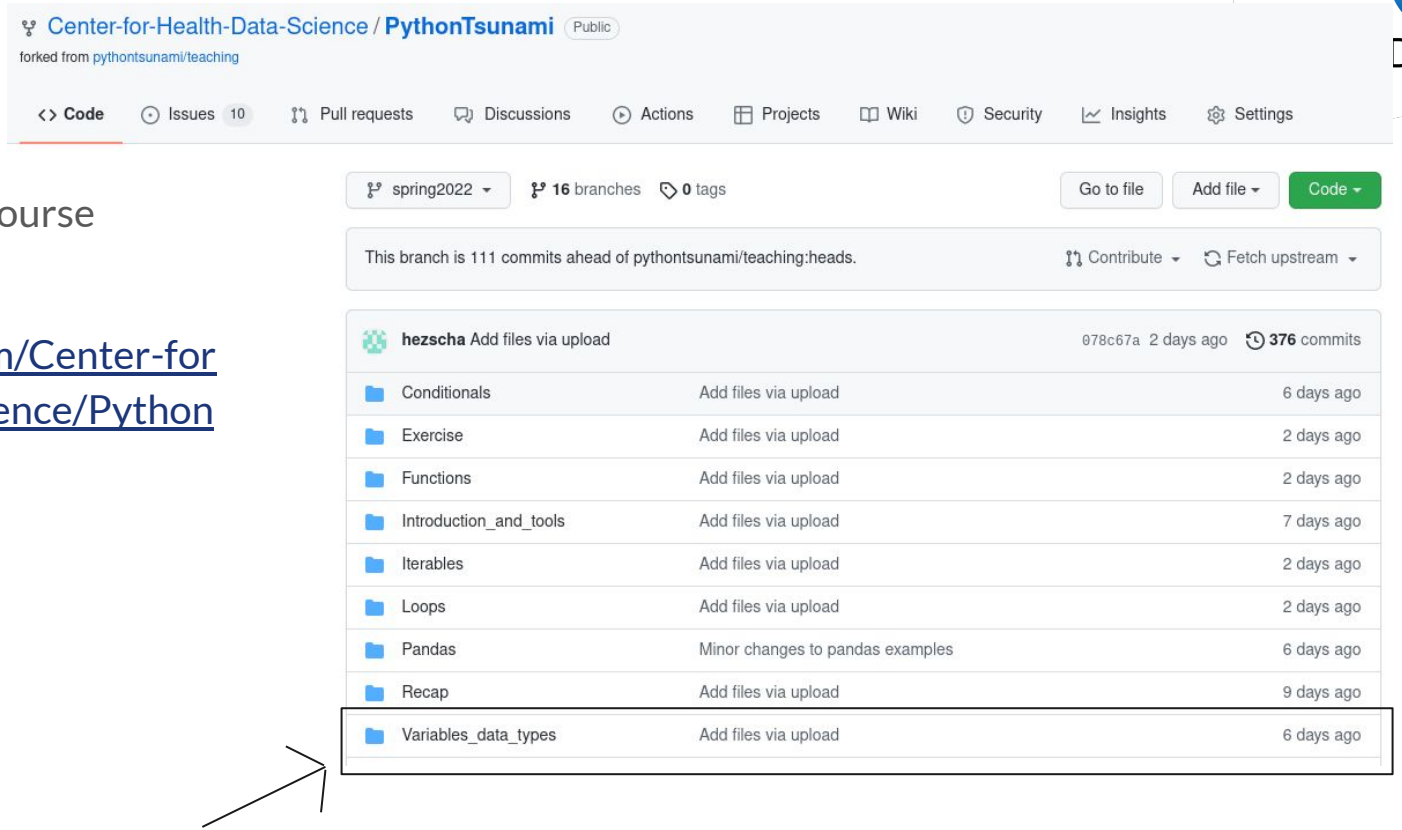
- tool to write, execute and share Python code through the browser
- requires no setup to use and provides free access to computing resources on Google's servers including GPUs
- is connected to a Google account and data and notebooks can be accessed through Google Drive.

We'll use Colab during the course.

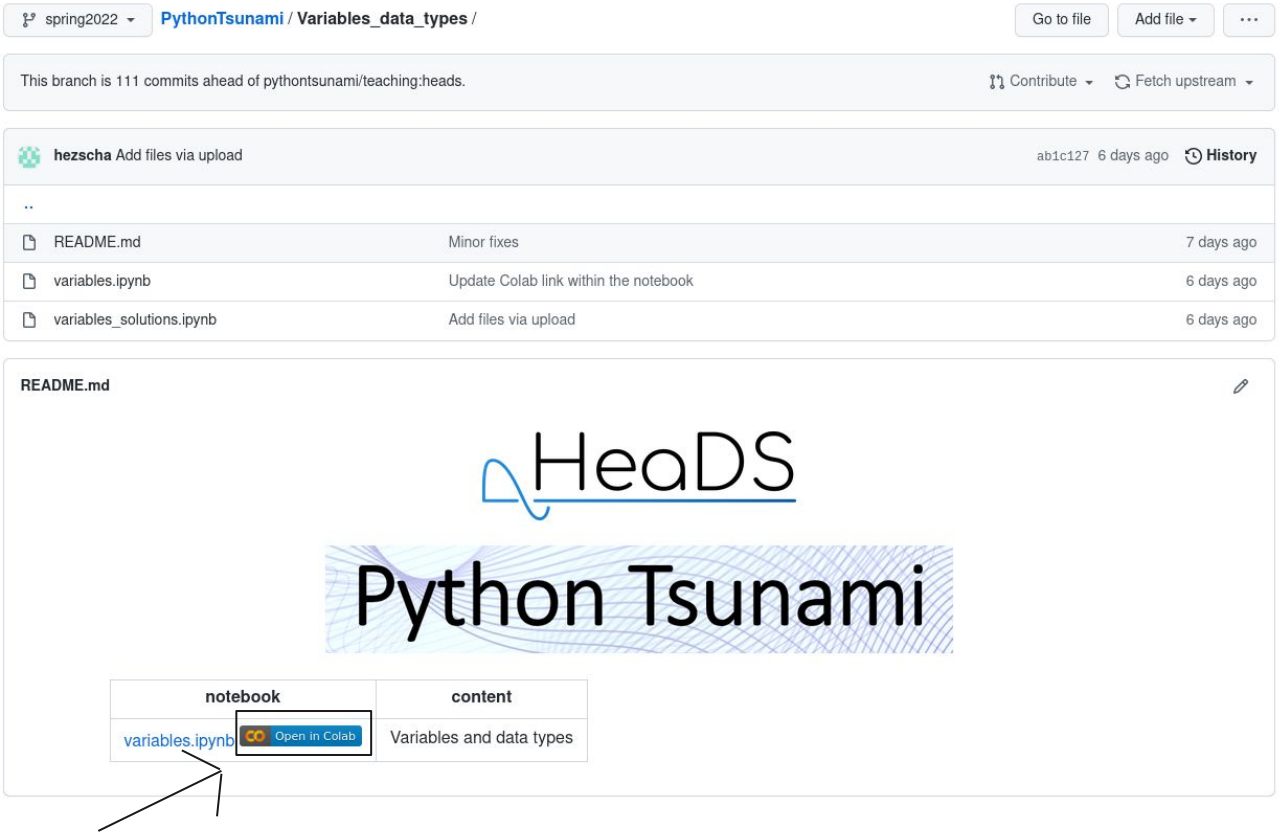
Course material

You can find the course material here:

https://github.com/Center-for-Health-Data-Science/PythonTsunami

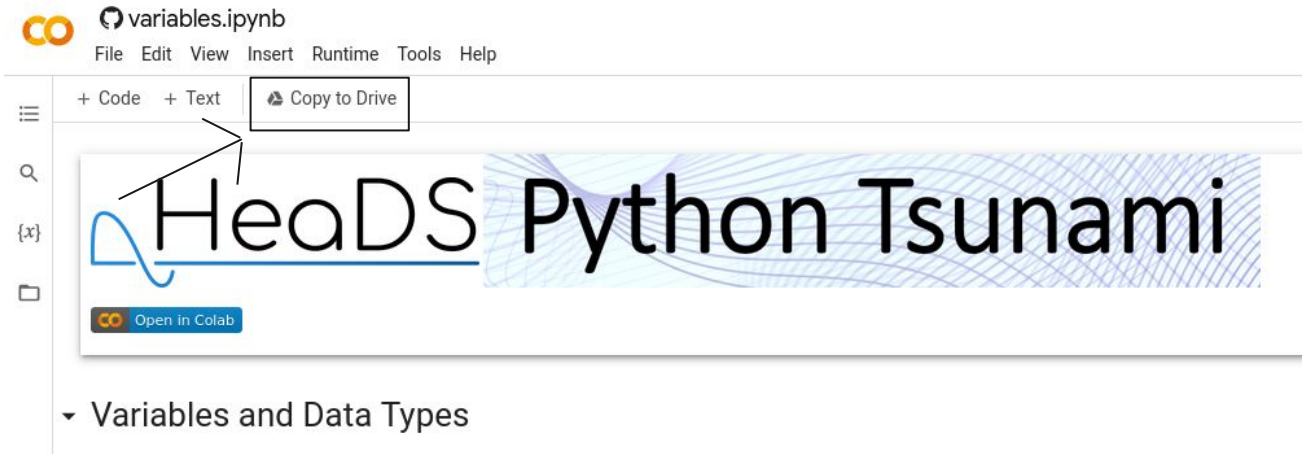


Course material



Course material

Remember to **save a copy** to your own google drive so you can save your notes and exercises!



Short Introduction

Take the next 5 mins to introduce yourself at your table:

- Name
- Position
- Unit
- What you do (very briefly!)

Using libraries/packages

Libraries

Python has many libraries, also called packages, that other programmers have developed. Find and **use** them!

Well-maintained libraries generally are:

- Tested
- Optimized
- Documented

There is no need to reinvent the wheel. During this course we will use:

- Pandas (all the data analysis!)
- Math (basic math)
- Plotly express (visualization)

Libraries

If you are running Python from a local installation, you need to have libraries **installed** before you can use them.

On Google Colab you can generally just import, they are already installed.

- Import the math library:

```
import math
```

- Now I can use functions from that library, i.e. calculating the logarithm or square root:

```
math.log(3)

math.sqrt(4)
```

Objects, variables, references

The language of python

Python is an **object-oriented** programming language. Therefore it helps if you primarily think of two different things:

Objects and functions*

- Objects are **pieces of information** (i.e. a number, a string of letters, a data table).
- We perform functions on objects. They are what we **do** to the information pieces.
- Which functions we can perform depends on the type of the object.

*technically functions are objects too, but let's not get too technical

An example

An object:

```
my_int = 3
```

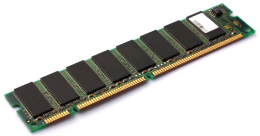
A function:

```
math.log(my_int)
```

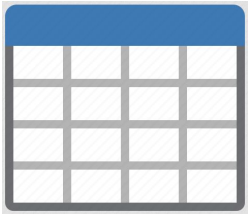
This does not work because the object is the wrong type:

```
math.log('hello')
```

Variables and Objects



RAM = physical memory



Object = piece of info, i.e. a pandas dataframe

```
my_df = read.csv(...)

my_int = 3

cities_list = ['New York',
'Copenhagen', 'Berlin']
```

Variable inside Python program

Every piece of data you use in Python is stored somewhere in physical memory as an **object.**

Variables are **references** to the stored object.