

Introduktion till Python - Facit

Filip Edström

Uppgift 1 - grunder och syntax

Tilldelning

Python

```
# Assigning a value to a variable  
x = 10  
y = "hello"
```

Datastrukturer och indexering

Python

```
# List  
v = [1, 2, 3]  
print(v)  
# List  
l = ["Apple", "Orange", "Pear"]  
print(l)  
# Dictionary  
d = {"apple": "crunchy", "orange": "juicy", "pear": "sweet"}  
print(d)
```

```
[1, 2, 3]  
['Apple', 'Orange', 'Pear']  
{'apple': 'crunchy', 'orange': 'juicy', 'pear': 'sweet'}
```

Loopar och if/else-satser

Python

```
numbers = range(1, 10)
for number in numbers:
    print(number)
    if number == 5:
        print("Five")
```

```
1
2
3
4
5
Five
6
7
8
9
```

Funktioner

Python

```
def multiply(x, y):
    return x * y
multiply(2, 22)
```

```
44
```

Paket

Python

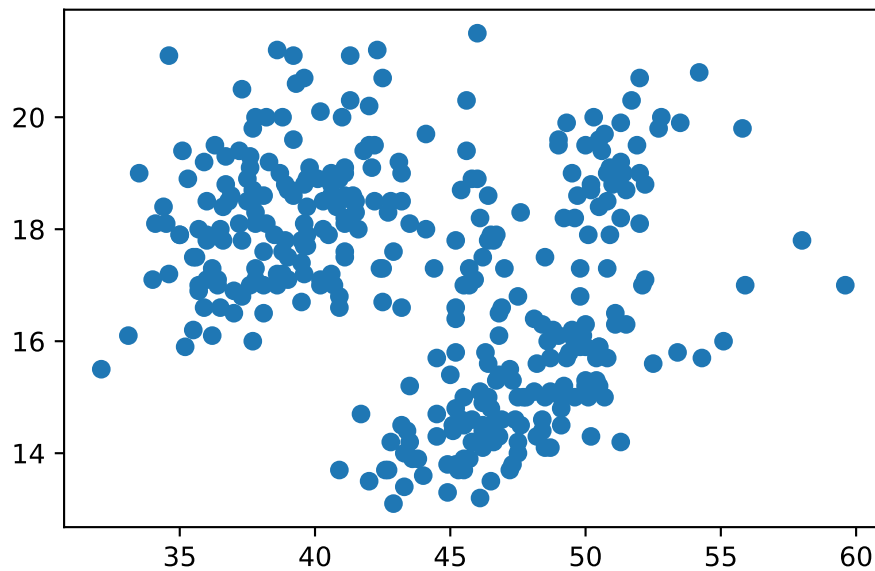
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

# Load data frame
```

```

penguins = pd.read_csv("penguins.csv")
# Print head of dataframe
penguins.head()
# Print head of species column
penguins["species"].head()
# Compute mean of flipper_length_mm
np.mean(penguins["flipper_length_mm"])
# Plot bill_length_mm vs bill_depth_mm
plt.scatter(penguins["bill_length_mm"], penguins["bill_depth_mm"])
plt.show()

```



Objektorienterad programmering

Python

```

class Animal:
    def __init__(self, name):
        self.name = name

    def speak(self):
        raise NotImplementedError("Subclass must implement this method")

class Dog(Animal):

```

```
def speak(self):
    return f"{self.name} says Woof!"

class Cat(Animal):
    def speak(self):
        return f"{self.name} says Meow!"

# Demonstrating Polymorphism
animals = [Dog("Buddy"), Cat("Whiskers")]

for animal in animals:
    print(animal.speak())
```

Buddy says Woof!
Whiskers says Meow!

Uppgift 2 - Centrala gränsvärdessatsen

Not implemented yet.