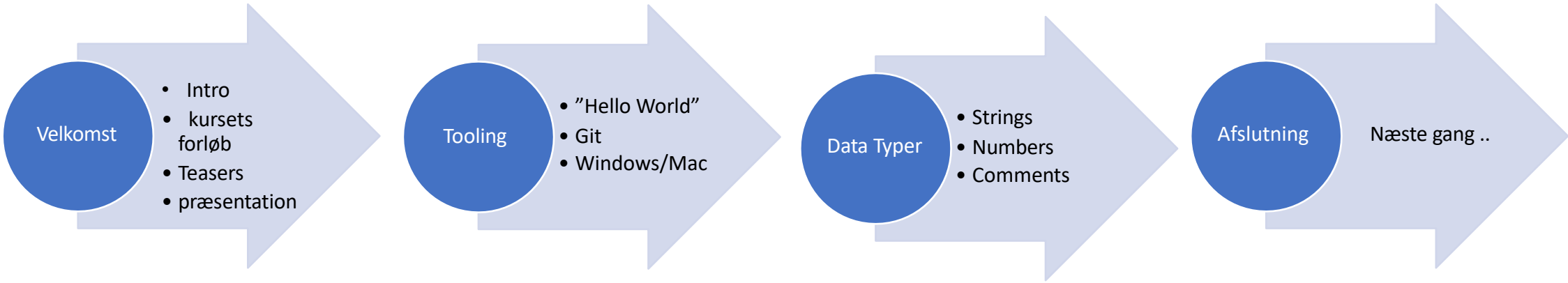
EVU Python

Program


Velkomst
• Intro
• kursets forløb
• Teasers
• præsentation
Tooling
• ”Hello World”
• Git
• Windows/Mac
Data Typer
• Strings
• Numbers
• Comments
Afslutning
Næste gang ..

Velkomst

kursets forløb

PART I: BASICS 24/8

Chapter 1: Getting Started 24/8

Chapter 2: Variables and Simple Data Types 24/8

Chapter 3: Introducing Lists 31/8

Chapter 4: Working with Lists 31/8

Chapter 5: if Statements 07/9

Chapter 6: Dictionaries. 07/9

Chapter 7: User Input and while Loops 14/9

Chapter 8: Functions 14/9

Chapter 9: Classes. 21/9

Chapter 10: Files and Exceptions. 21/9

Chapter 11: Testing Your Code 21/9

PART II: PROJECTS 28/9

Project 1: Alien Invasion 28/9

Chapter 12: A Ship That Fires Bullets 28/9

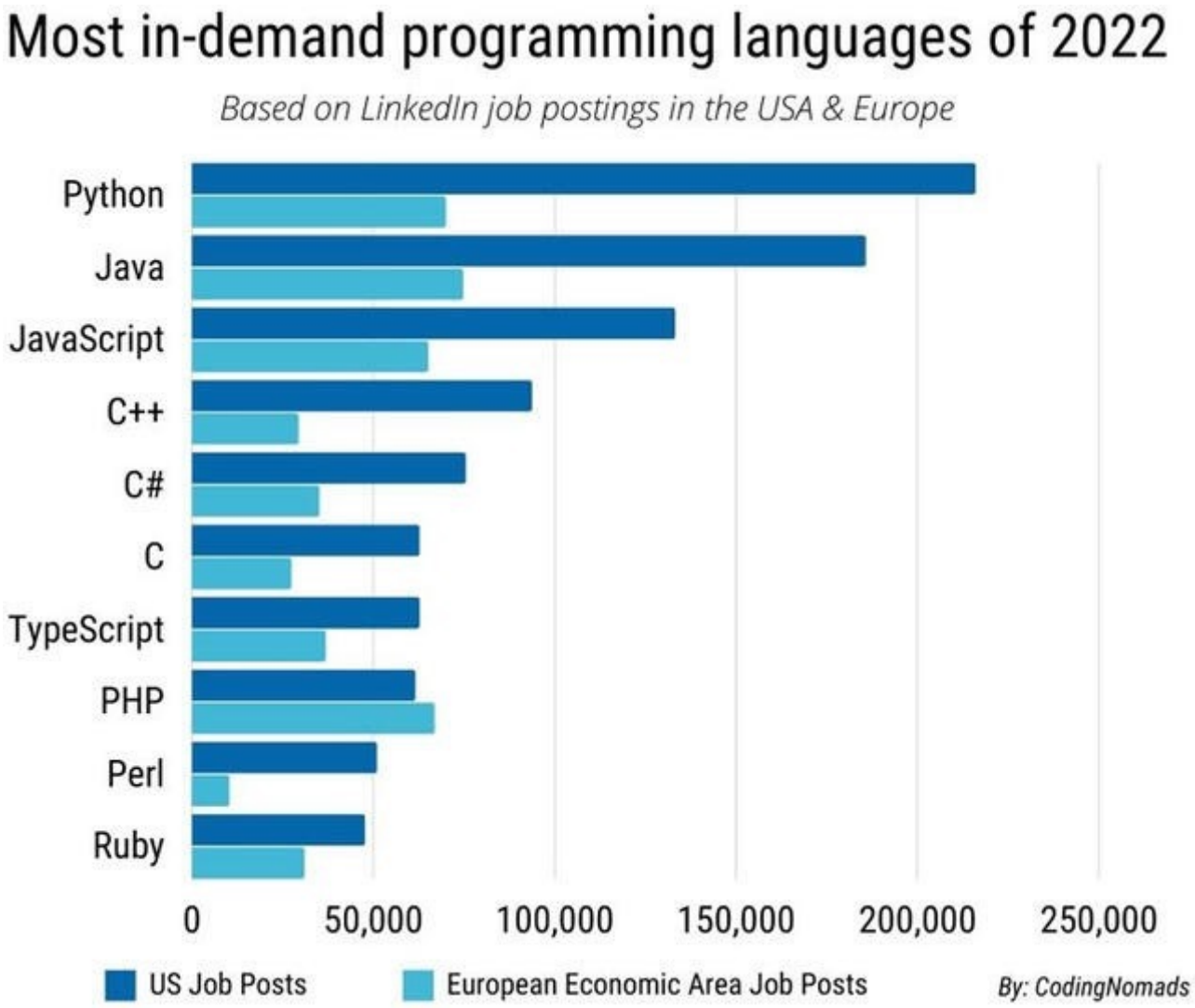
Project 2: Kill the birds 5/10

Repetition 5/10

Her forlader vi bogen

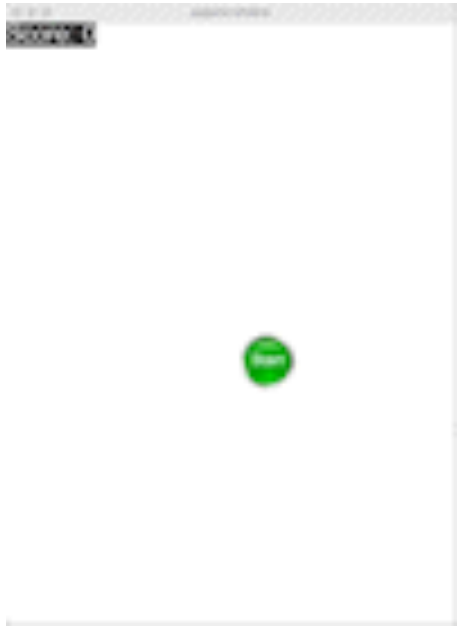
Python

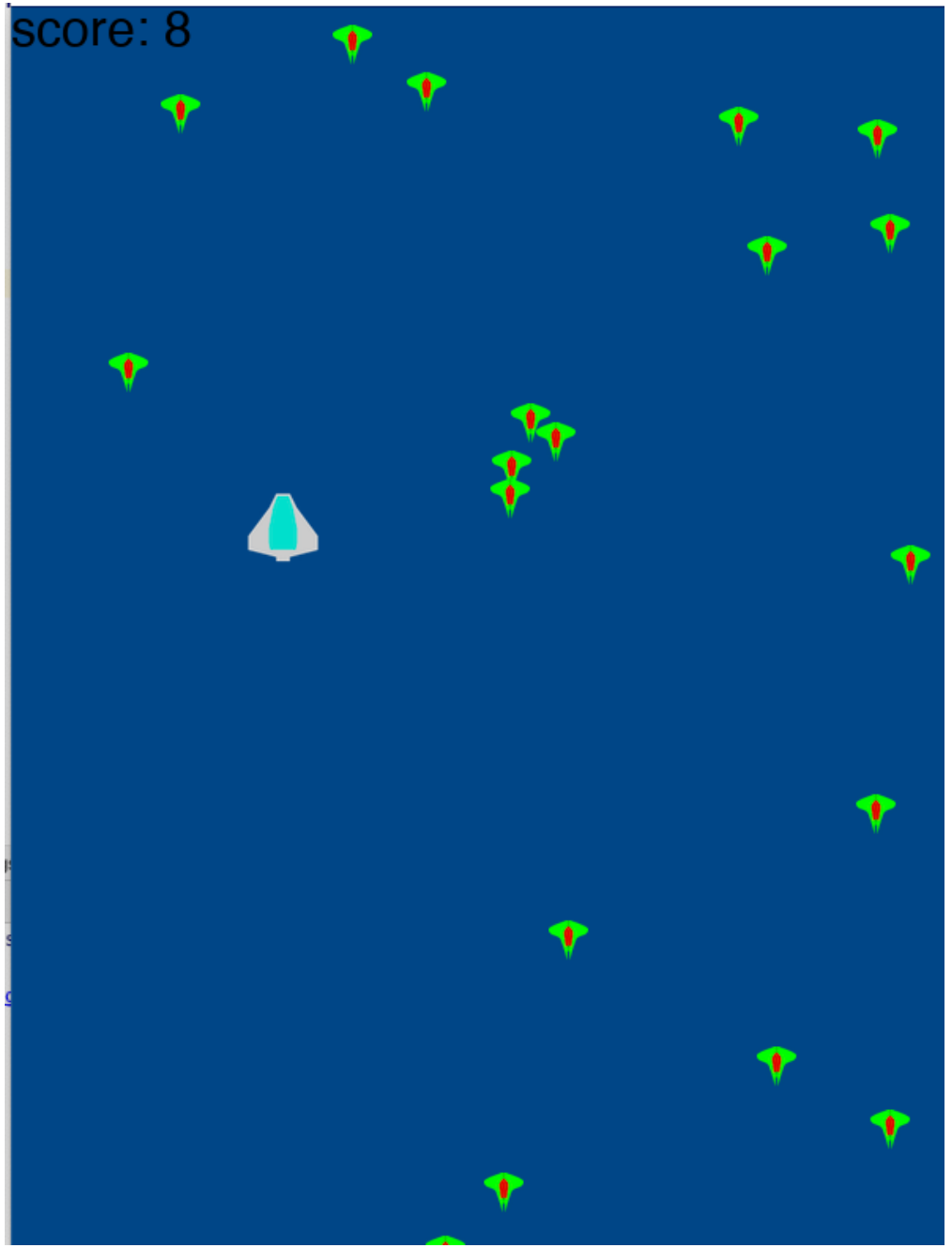
Stats



Pygame

Final pygame





The way

Skovlen ..

I/O

Loops

Operators

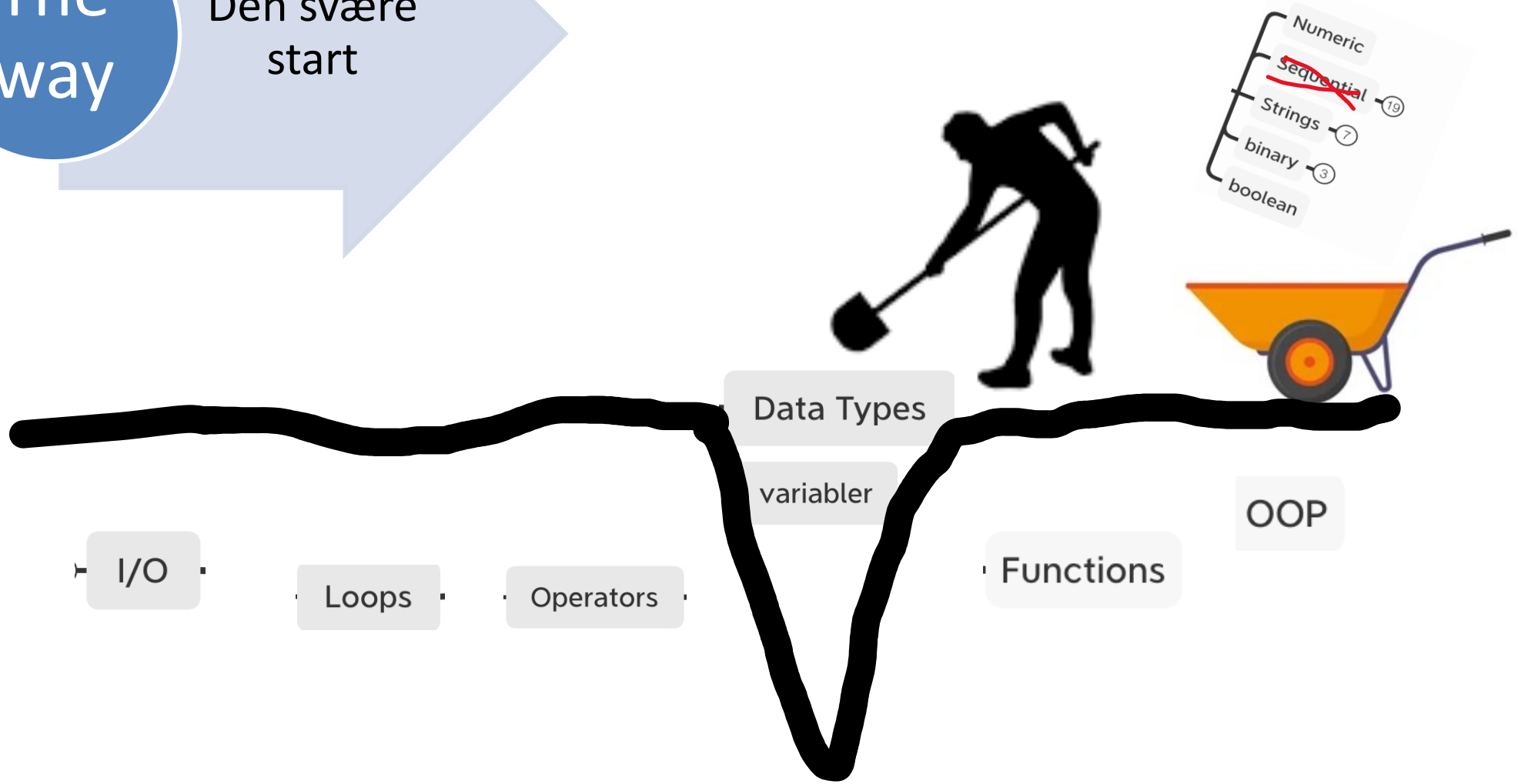
Data Types

Functions

OOP

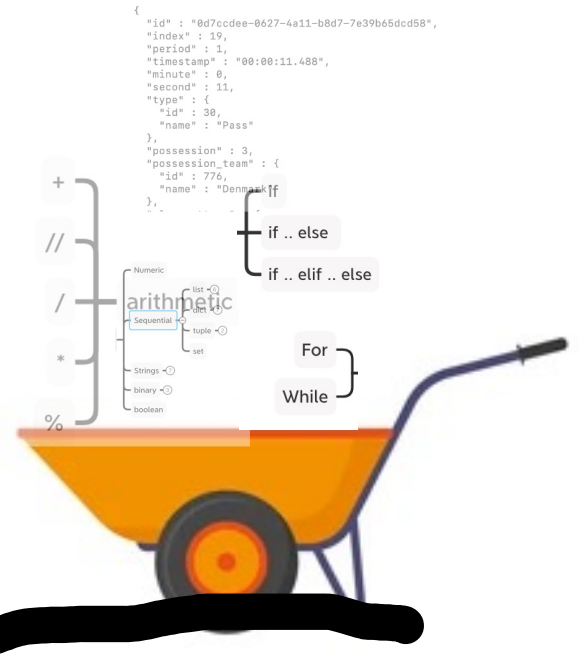
The way

Den svære start



The way

Så sker der noget ..



I/O

Loops

Operators

variabler

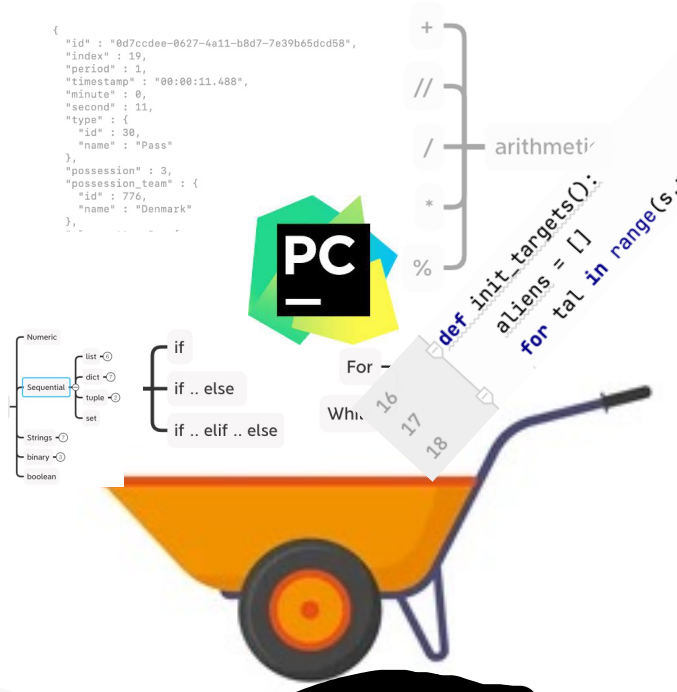
Data Types

Functions

OOP

The way

Orden i kaos …



I/O

Loops

Operators

variabler

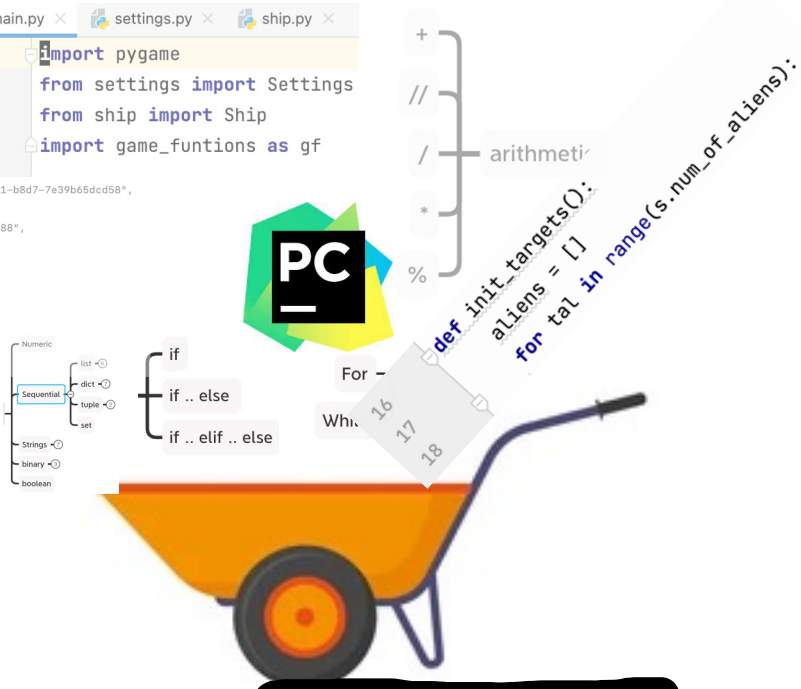
Data Types

Functions

OOP

The way

Klar til Game Dev



OOP

Operators

Data Types

Loops

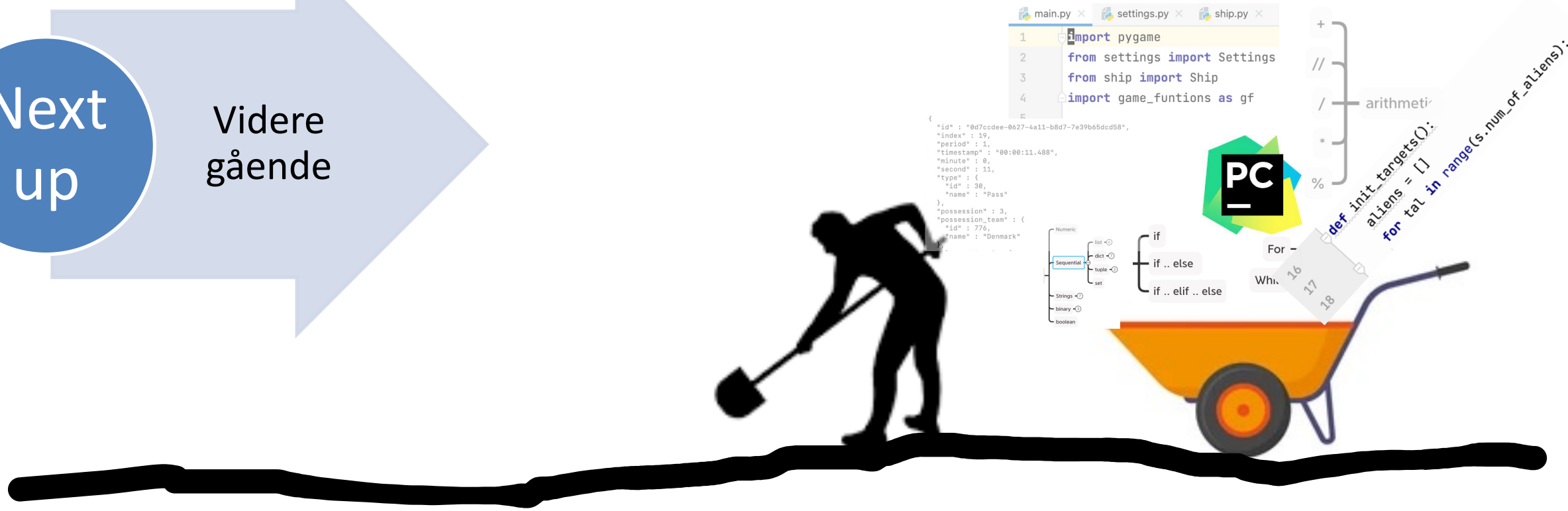
variabler

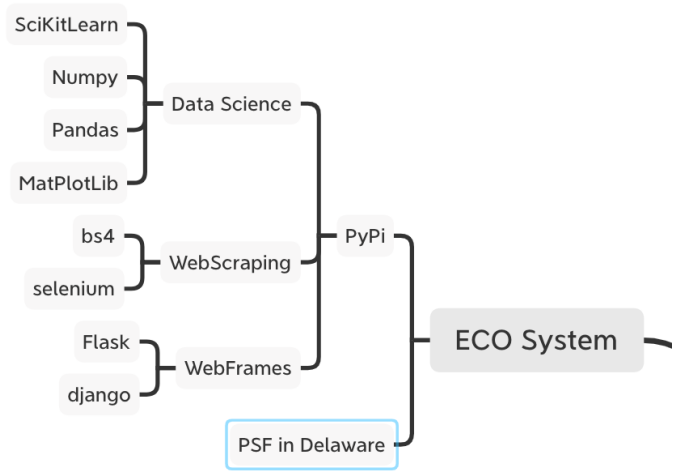
Functions

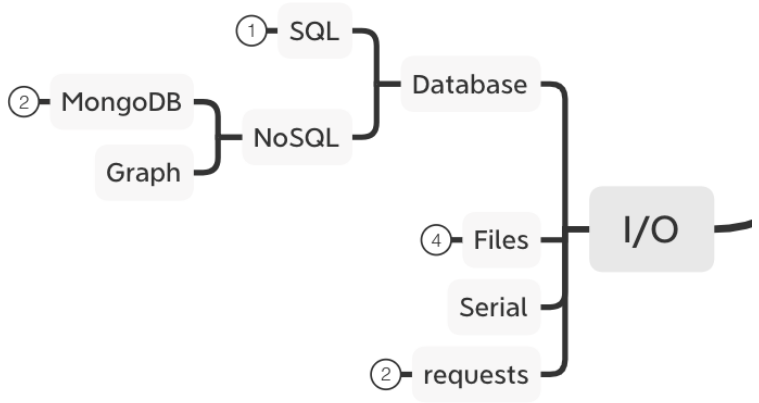
I/O

Next up

Videre gående



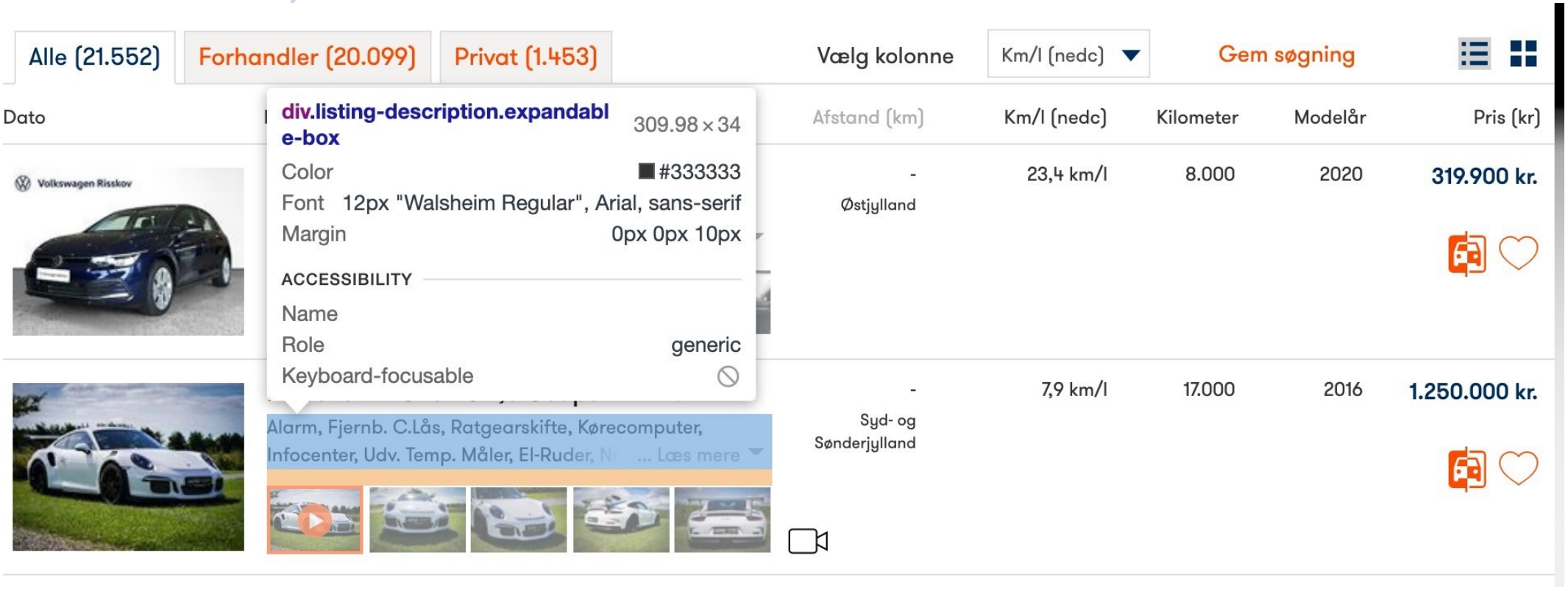




PAUSE

Teaser

Web Scraping

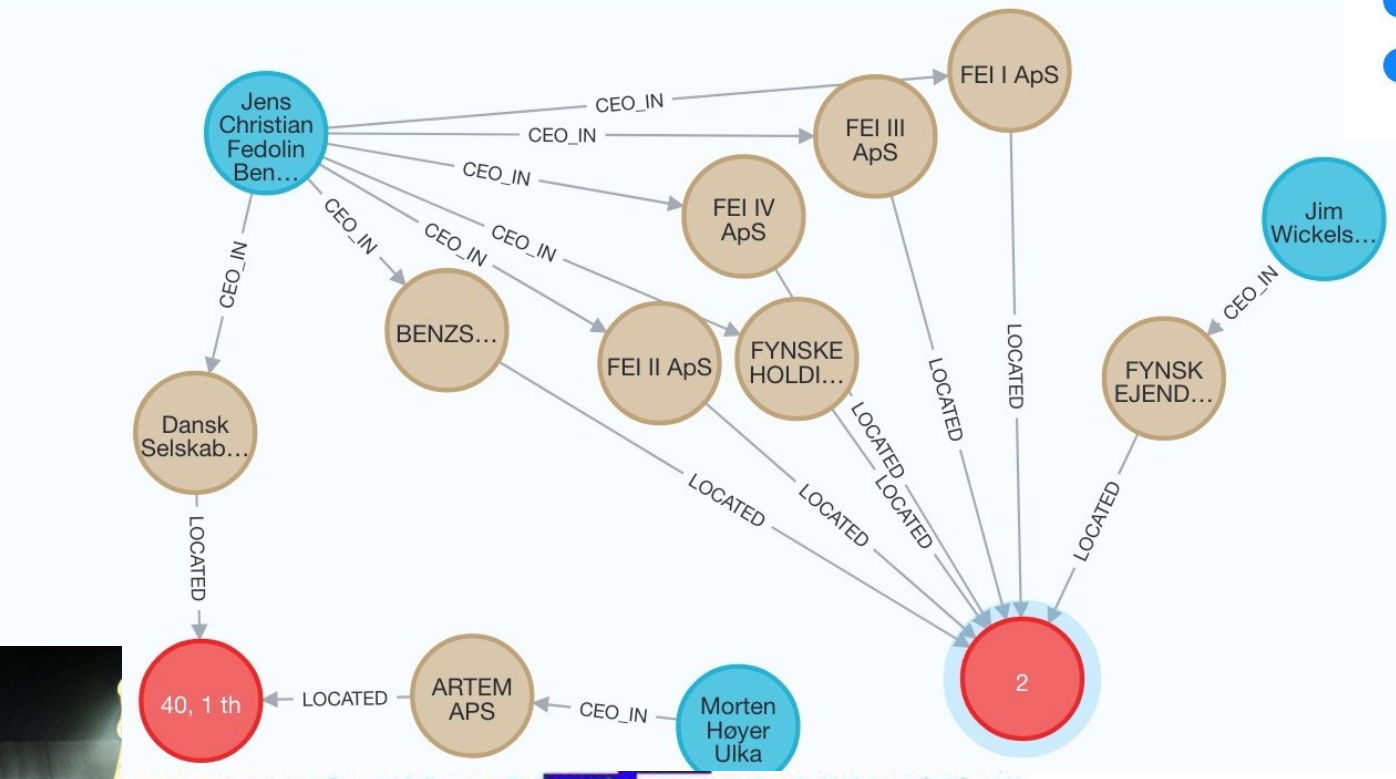


Teaser

Fraud Detection

```
neo4j$ match (n:Location) where id(n)=55 return n
```

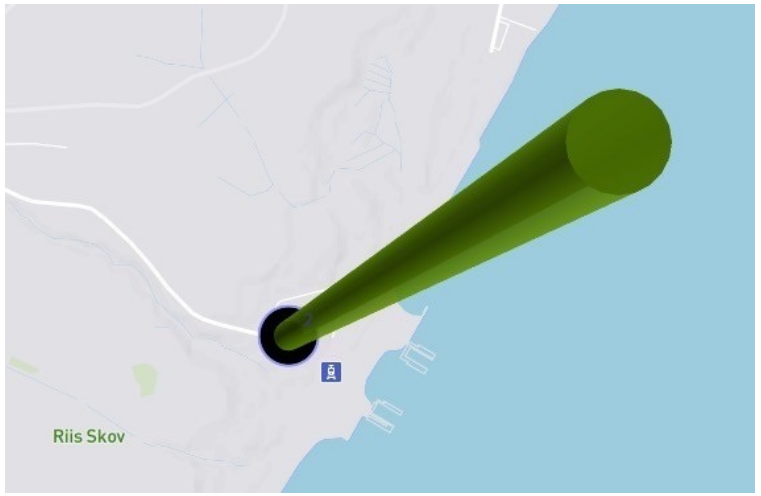
Graph
Table
Text
Code

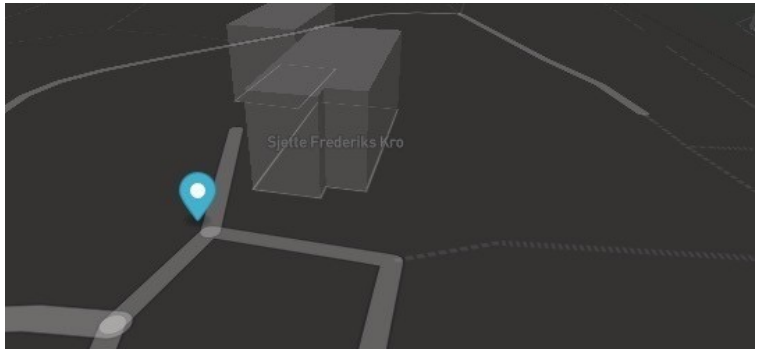


```
c174">9700</c:AddressOfAuditorPostCodeIdentifier>
Brønderslev</c:AddressOfAuditorDistrictName>
rdjyske Bank</c:NameOfFinancialInstitution>
Vodskovvej</c:AddressOfFinancialStreetName>
extRef="c174">43</c:AddressOfFinancialStreetBuildingIdentifier>
="c174">9310</c:AddressOfFinancialPostCodeIdentifier>
'>Vodskov</c:AddressOfFinancialDistrictName>
ester Hassing</e:PlaceOfSignatureOfStatement>
2017-11-07</e:DateOfApprovalOfAnnualReport>
ef="c637">Poul Erik Madsen</d:NameAndSurnameOfMemberOfExecutiveBoard>
tRef="c646">Poul Erik Madsen</d:NameAndSurnameOfMemberOfSupervisoryBoard>
tRef="c647">Martin Nørgaard</d:NameAndSurnameOfMemberOfSupervisoryBoard>
<f:SignatureOfAuditorsPlace contextRef="c174">Brønderslev</f:SignatureOfAuditorsPlace>
```

Teaser

REST API





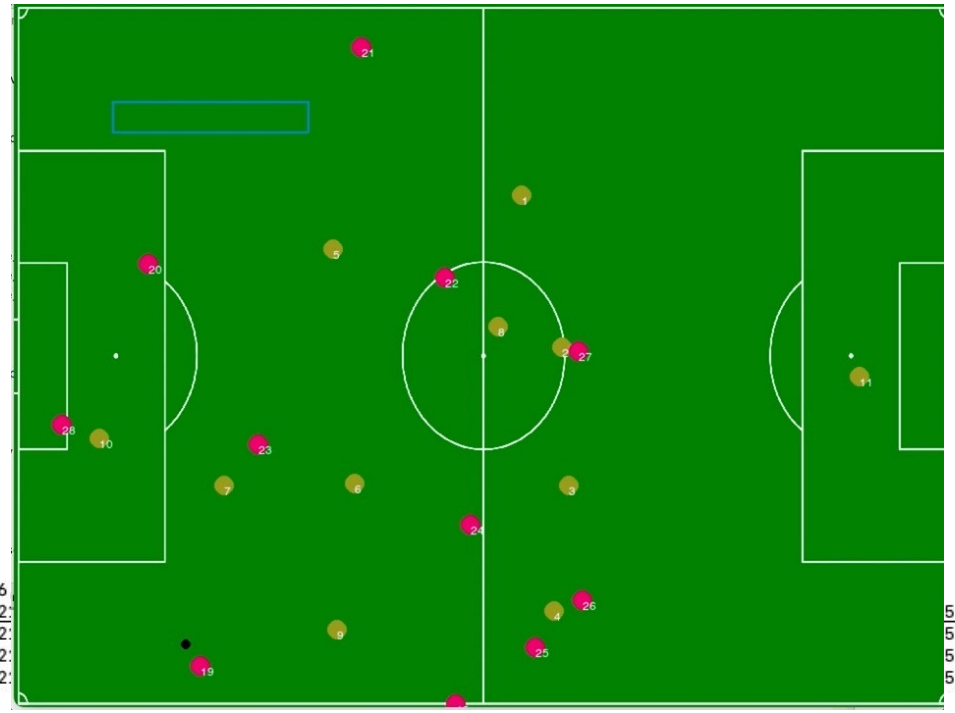
http shell javascript

```
GET /devices HTTP/1.1
Authorization: Bearer {BEARER_TOKEN}
Host: api.cityflow.live
```


```
{
  id: e00fce689f02a96799f34fc2,
  type: 150,
  location: 190,
  latitude: 56.1770897,
  longitude: 10.2296247,
  location_name: Salonvejen,
  city: Risskov,
  country: Denmark,
  roles: [
    4
  ],
  permissions: [],
  tags: [
    Risskov
  ]
},
```

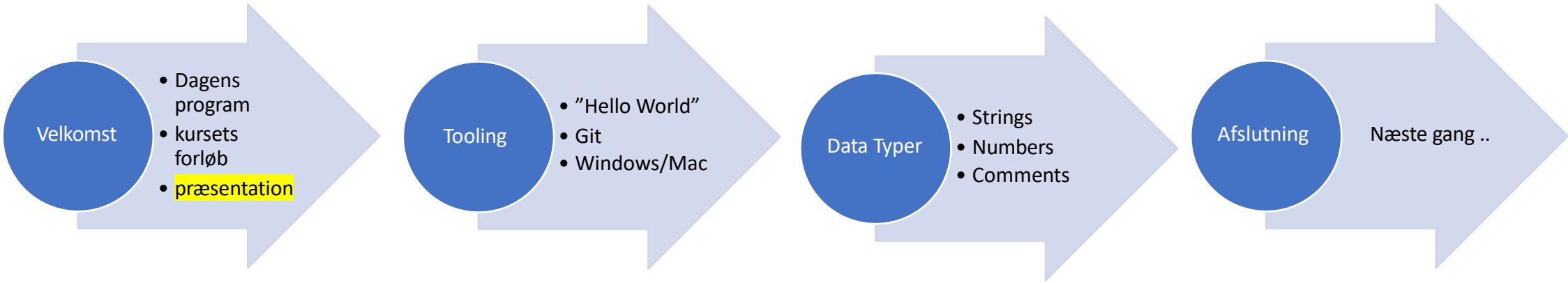
Teaser

Sports Analytics



```
Period,Frame,Time [s],Player11,,Player1,,Player2,,Player3,,Player4,,Player5,,Player6,
1,1,0.04,0.00082,0.48238,0.32648,0.65322,0.33701,0.48863,0.30927,0.35529,0.32137,0.2[illegible]5243,0.43269,NaN,NaN,NaN,NaN,NaN,NaN,0.45472,0.38709
1,2,0.08,0.00096,0.48238,0.32648,0.65322,0.33701,0.48863,0.30927,0.35529,0.32137,0.2[illegible]5243,0.43269,NaN,NaN,NaN,NaN,NaN,NaN,0.49645,0.40656
1,3,0.12,0.00114,0.48238,0.32648,0.65322,0.33701,0.48863,0.30927,0.35529,0.32137,0.2[illegible]5243,0.43269,NaN,NaN,NaN,NaN,NaN,NaN,0.53716,0.42556
1,4,0.16,0.00121,0.48238,0.32622,0.65317,0.33687,0.48988,0.30944,0.35554,0.32142,0.2[illegible]55236,0.43313,NaN,NaN,NaN,NaN,NaN,NaN,0.55346,0.42231
```

Program


Velkomst
• Dagens program
• kursets forløb
• præsentation
Tooling
• ”Hello World”
• Git
• Windows/Mac
Data Typer
• Strings
• Numbers
• Comments
Afslutning
Næste gang ..

Øvelse: Dan grupper ud fra …

	1	2+5	3	4+6
gruppe 1				
gruppe 2				
gruppe 3				
gruppe 4				
gruppe 5				
gruppe 6				

Øvelse

Opgave 2-5: Bliv enige i gruppen om hvem der har fundet det bedste citat og præsentér løsningen
Opgave 2-3: Hvert medlem præsenteres i en notebook-celle:

```
print(navn+" arbejder på "+arbplads+ " som "+rolle.lower()+ "der ligger i "+location.title())
```

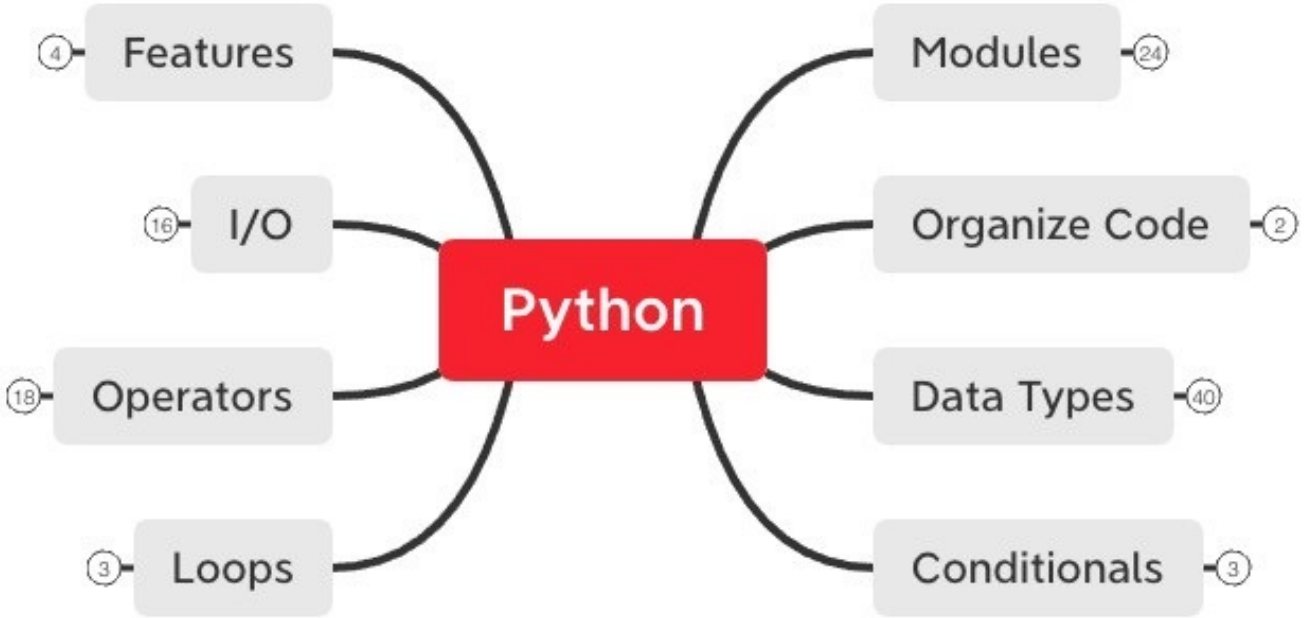
```
Thorbjørn Wulf arbejder på CPH Business som underviserder ligger i Lyngby
```

(og hvis man er i karriere-skift udfylder man med drømme - job, -rolle og –location)

ekstraopgave: Find de 5 forskellige måder man kan *concatenate strings* på i python og illustrer med et simpelt eksempel

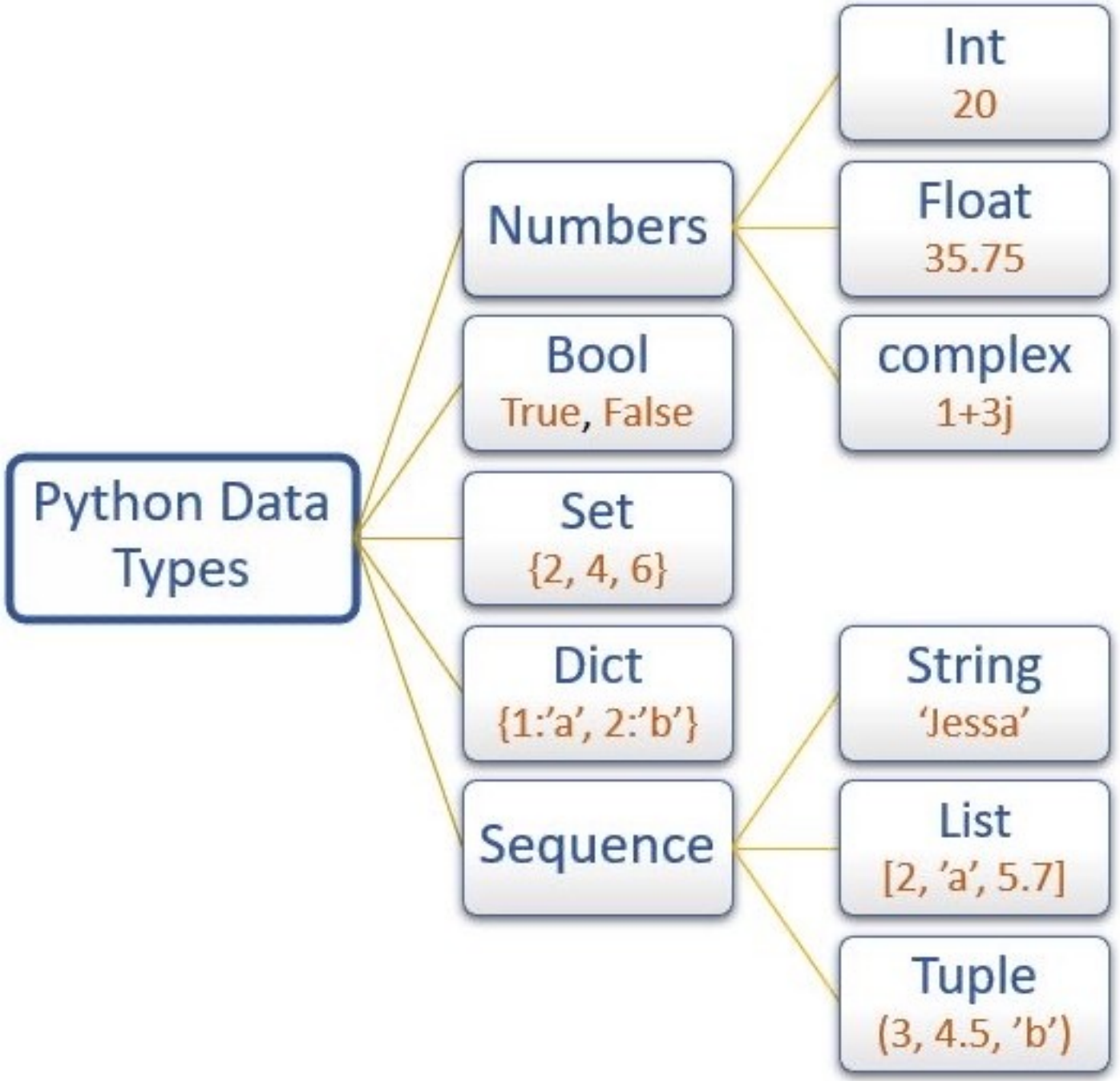
Pause

Python Struktur



Datatyper

”Simple”

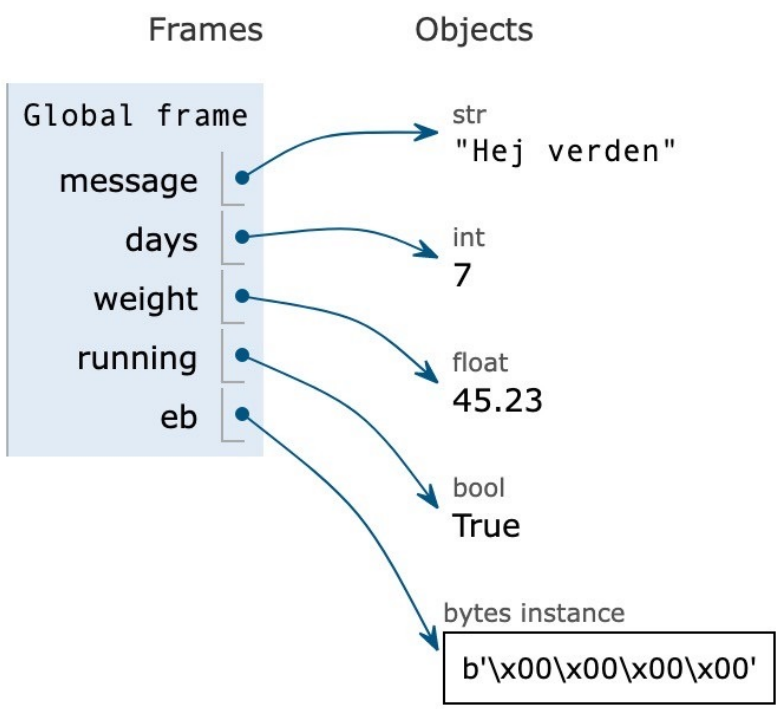


A Python variable is a reserved memory location to store values.

Every value in Python has a datatype.

Datatyper

”Simple”



A Python variable is a reserved memory location to store values. Every value in Python has a datatype.

Text Type:	str
Numeric Types:	int, float, complex
Sequence Types:	list, tuple, range
Mapping Type:	dict
Set Types:	set, frozenset
Boolean Type:	bool
Binary Types:	bytes, bytearray, memoryview

When you’re using variables in Python, you need to adhere to a few rules and guidelines. Breaking some of these rules will cause errors -> case CVR:

```
2talRevision
2talRevision Reg
3 H Revision
3 H Revision
```

Variabler

Avoid Keywords

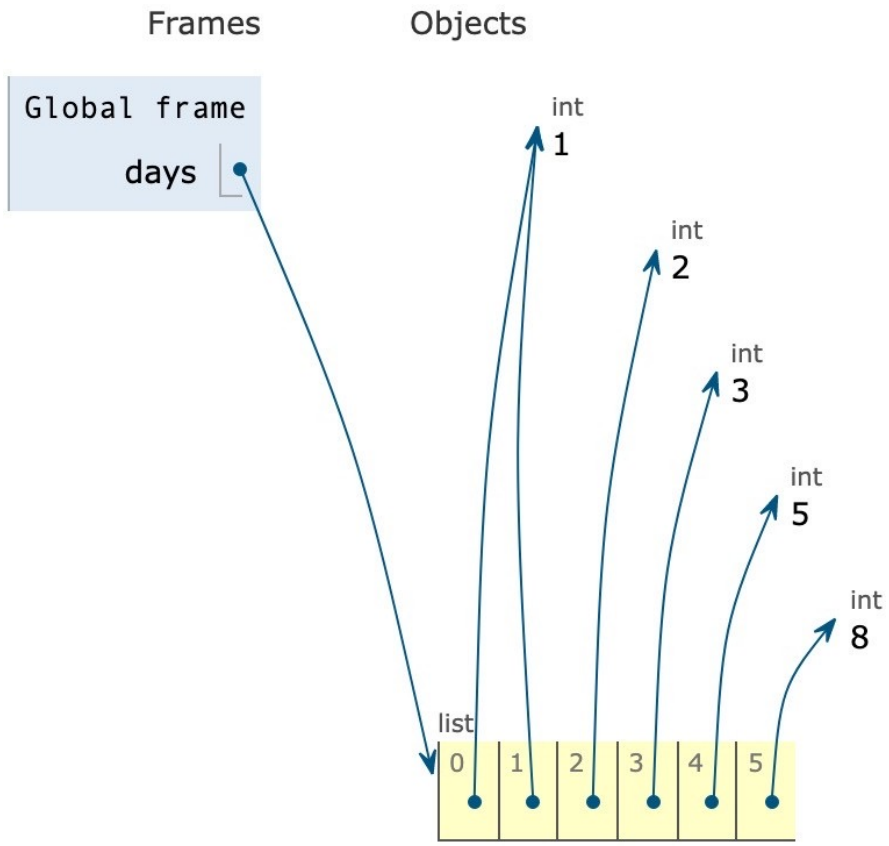
Python Keywords

Each of the following keywords has a specific meaning, and you'll see an error if you try to use them as a variable name.

`False`	`class`	`finally`	`is`	`return`
`None`	`continue`	`for`	`lambda`	`try`
`True`	`def`	`from`	`nonlocal`	`while`
`and`	`del`	`global`	`not`	`with`
`as`	`elif`	`if`	`or`	`yield`
`assert`	`else`	`import`	`pass`	
`break`	`except`	`in`	`raise`	

Datatyper

”Komplekse"



Text Type:	`str`
Numeric Types:	`int`, `float`, `complex`
Sequence Types:	`list`, `tuple`, `range`
Mapping Type:	`dict`
Set Types:	`set`, `frozenset`
Boolean Type:	`bool`
Binary Types:	`bytes`, `bytearray`, `memoryview`

Variabler

How to?

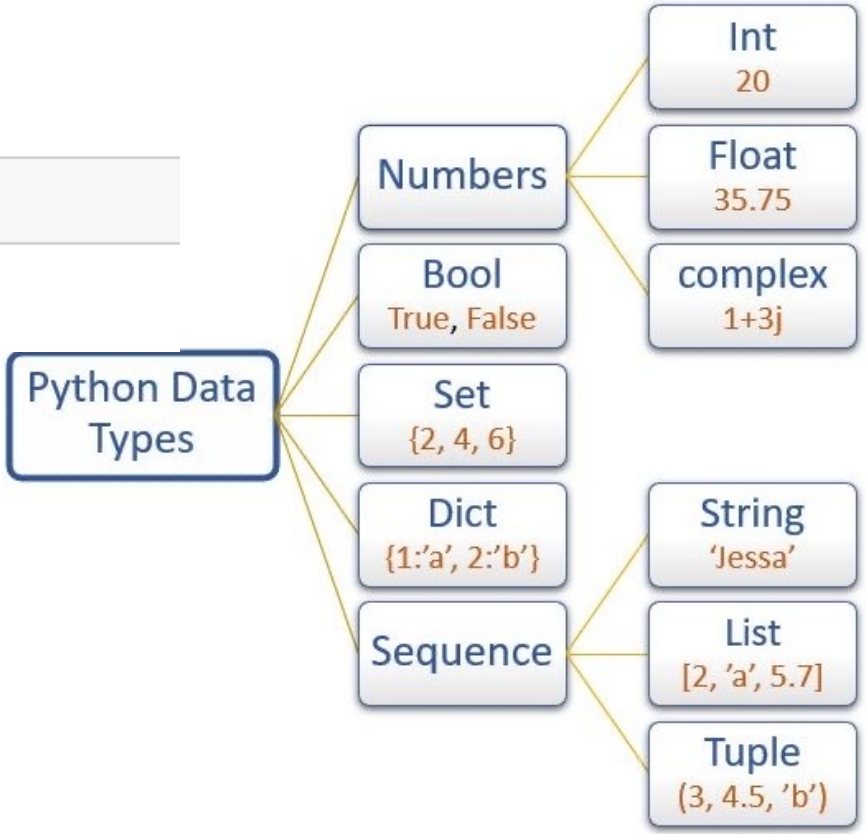
notebook

Dynamic Typing

Python uses *dynamic typing*, meaning you can reassign variables to different data types. This makes Python very flexible in assigning data types; it differs from other languages that are *statically typed*.

```
In [7]: type("Jessa")
Out[7]: str
```

```
In [8]: type( )
Out[8]: int
```



Variabler

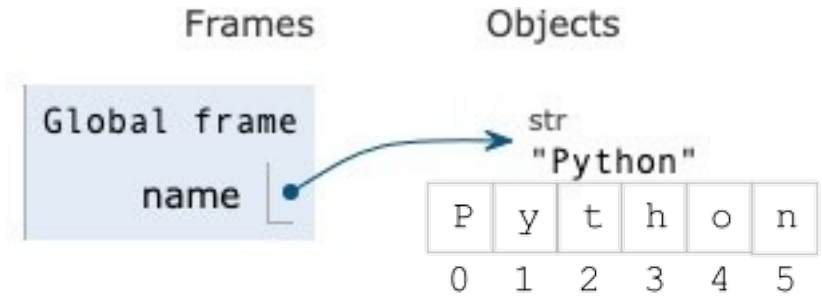
Strings

Definition

A string is simply a series of characters (” ” or ‘ ‘)

”*sequence of bytes representing unicode characters*”

Memory



Standard

print(”kurt verner”)

Special

print(”Kurt \t Verner”)

Speciel special

```
In [87]: navn="BMW \x99"

In [88]: print(navn)
BMW ™
```

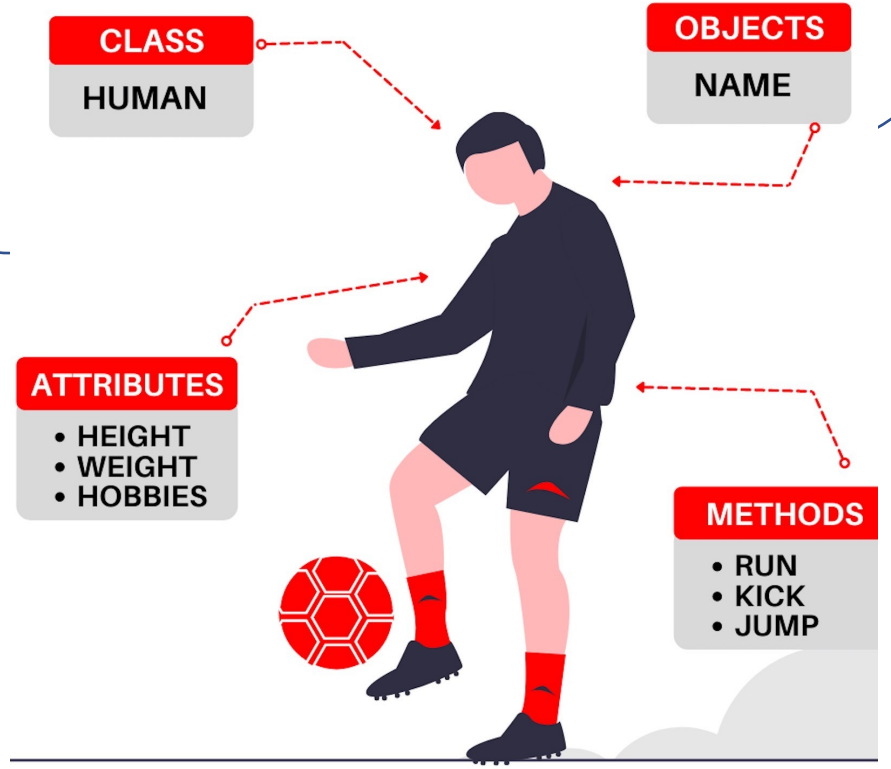
```
In [104]: print(navn)
          McDonalds ©

In [106]: print(pris)
          200 £

In [115]: print(t)
          ½ + ¼ = ¾
```

https://kb.digital-detective.net/display/BF/Extended+ASCII+Table

Variabler

Strings

The Dot-notation




```
messi.run(12)
messi.jump()
....

”messi”.toupper()
”messi”.strip()
```

Variabler

Strings

Common built-in functions

print(”kurt Verner”)

len(”kurt Verner”)

Common methods

Function	Description
format()	It's used to create a formatted string from the template string and the supplied values.
split()	Python string split() function is used to split a string into the list of strings based on a delimiter.
join()	This function returns a new string that is the concatenation of the strings in iterable with string object as a delimiter.
strip()	Used to trim whitespaces from the string object.
format_map()	Python string format_map() function returns a formatted version of the string using substitutions from the mapping provided.
upper()	We can convert a string to uppercase in Python using str.upper() function.
lower()	This function creates a new string in lowercase.
replace()	Python string replace() function is used to create a new string by replacing some parts of another string.
find()	Python String find() method is used to find the index of a substring in a string.
translate()	Python String translate() function returns a new string with each character in the string replaced using the given translation table.

Strings

Øvelser

In [127]:
```
vej="Kurfürstendam"
nr = "12"
zip = 8000
city = "Århus"
print()
```

```
Kurfürstendam 12
8000 Århus
```

In [131]:
```
firma="Hvidkilde A/S "
# gem den rensede værdi i en ny variabel
print(f"Firmanavn: {firma}.")
```

```
Firmanavn: Hvidkilde A/S .
```

In [133]:
```
firma="Hvidkilde A/S"
# gem den rensede værdi i en ny variabel af samme navn
print(f"Firmanavn: {firma}.")
```

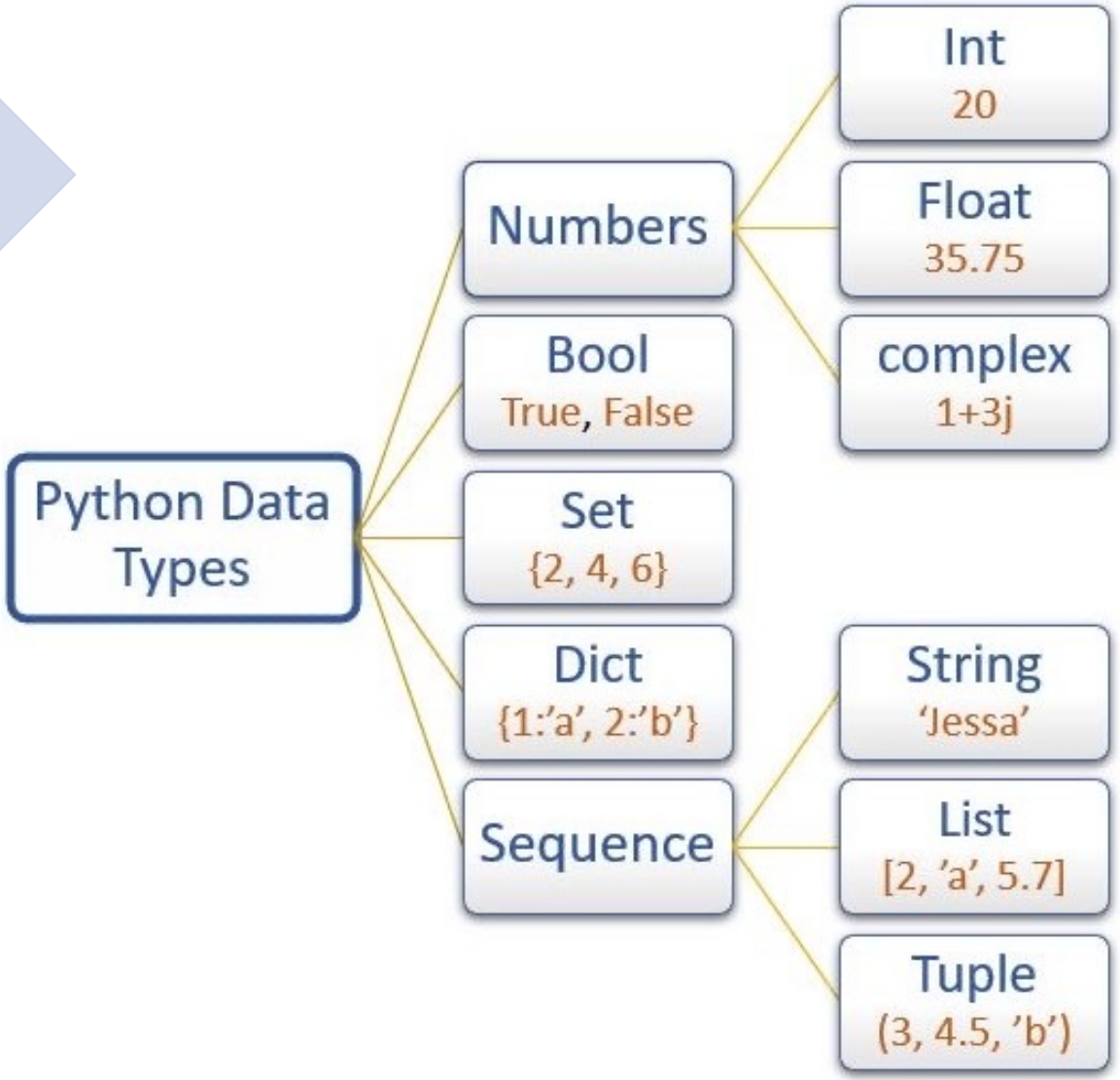
```
Firmanavn: Hvidkilde AS.
```

In [135]:
```
full_navn="Kurt Verner Hansen"
dele = full_navn.split()
```

Pause

Variabler

Numbers



Variabler

Numbers

- Int 20
- Float 35.75
- complex 1+3j

11	Left-to-right	* / %	Multiplication Division Modulo
10	Left-to-right	+ -	Addition Subtraction

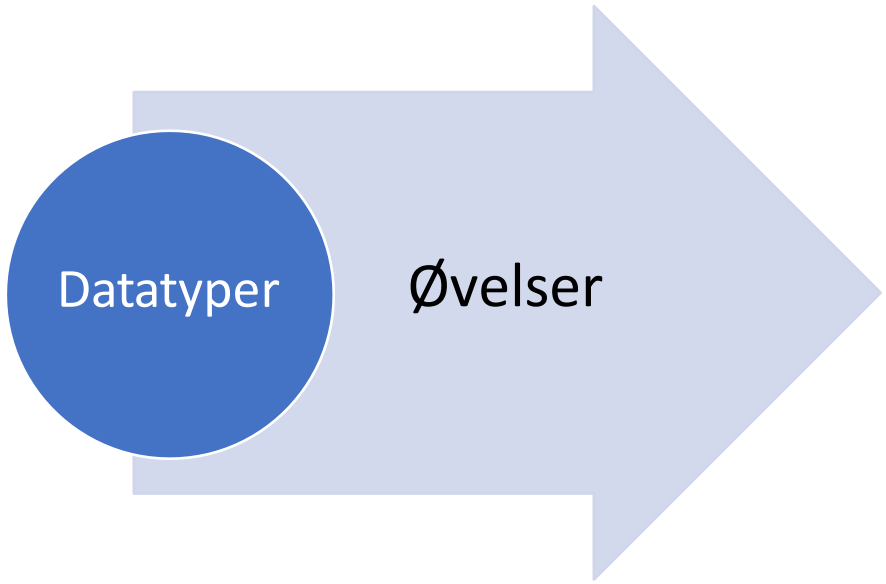
Functions

abs()	divmod()	input()	open()	staticmethod()
all()	enumerate()	int()	ord()	str()
any()	eval()	isinstance()	pow()	sum()
basestring()	execfile()	issubclass()	print()	super()
bin()	file()	iter()	property()	tuple()
bool()	filter()	len()	range()	type()
bytearray()	float()	list()	raw_input()	unichr()
callable()	format()	locals()	reduce()	unicode()
chr()	frozenset()	long()	reload()	vars()
classmethod()	getattr()	map()	repr()	xrange()
cmp()	globals()	max()	reversed()	zip()
compile()	hasattr()	memoryview()	round()	__import__()
complex()	hash()	min()	set()	apply()
delattr()	help()	next()	setattr()	buffer()
dict()	hex()	object()	slice()	coerce()
dir()	id()	oct()	sorted()	intern()

Data

Operators

Operator	Description
**	Exponentiation (raise to the power)
~, +, -	Complement, unary plus and minus (method names for the last two are +@ and -@)
*, /, %, //	Multiply, divide, modulo and floor division
+, -	Addition and subtraction
>>, <<	Right and left bitwise shift
&	Bitwise 'AND'
^	Bitwise exclusive 'OR'
|	Bitwise 'OR'
in, not in, is, is not, <, <=, >, >=, !=, ==	Comparison operators, equality operators, membership and identity operators
not	Boolean 'NOT'
and	Boolean 'AND'
or	Boolean 'OR'
=, %=, /=, //=, -=, +=, *=, **=	Assignment operators



- 2-8 – 8-tals stykkker (add,sub,mult og div)

Tjek dagens github for filen L2-Numbers-exercises

Afslutning

Næste gang ..

PART I: BASICS 24/8

Chapter 1: Getting Started 24/8

Chapter 2: Variables and Simple Data Types 24/8

Chapter 3: Introducing Lists 31/8

Chapter 4: Working with Lists 31/8

Chapter 5: if Statements 07/9

Chapter 6: Dictionaries. 07/9

Chapter 7: User Input and while Loops 14/9

Chapter 8: Functions 14/9

Chapter 9: Classes. 21/9

Chapter 10: Files and Exceptions. 21/9

Chapter 11: Testing Your Code 21/9

PART II: PROJECTS 28/9

Project 1: Alien Invasion

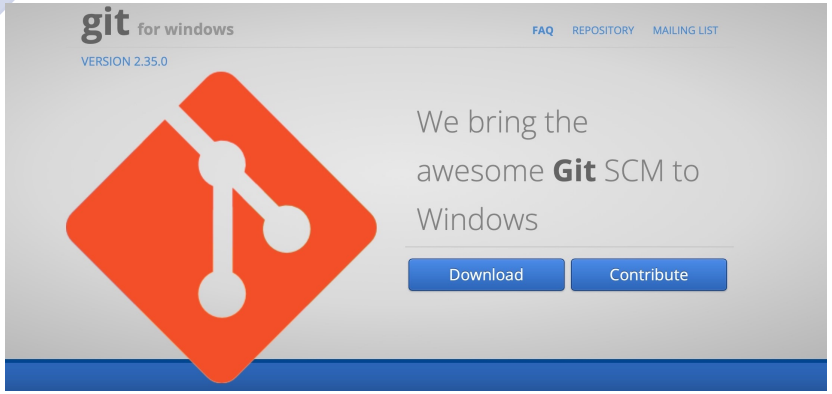
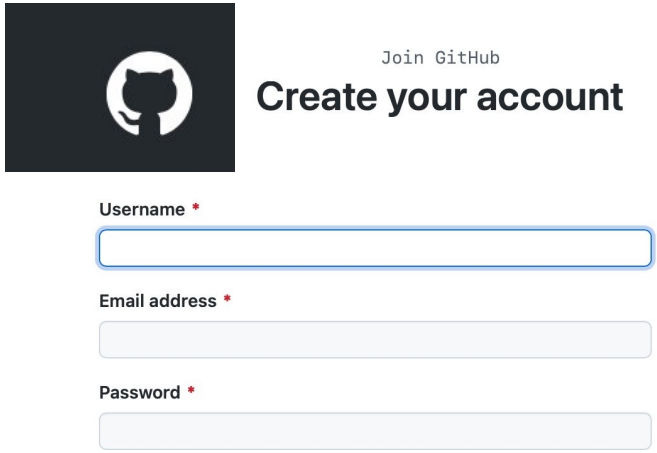
Chapter 12: A Ship That Fires Bullets 28/9

Project 2: Kill the birds 5/10

Repetition 5/10

Afslutning
Næste gang ..

Python

Tools

- Install GitBash (windows) or Git
- Create Git-account