

HTTP Status	
<b>Informative</b>	
100	Continue
101	Switching Protocol
<b>Success</b>	
200	OK
201	Created (POST)
202	Accepted
203	Non-Authoritative Information
204	No Content
<b>Redirect</b>	
301	Move Permanently
<b>Client Error</b>	
400	Bad Request
401	Unauthorized
402	Payment Required
403	Forbidden
404	Not Found
408	Timeout
429	Too Many Requests
451	Illegal
<b>Server Error</b>	
500	Internal Server Error
502	Bad Gateway
503	Service Unavailable

Python	
<b>List</b>	
lst	[10, 20, 30, 40, 50]
lst[:-1]	[10, 20, 30, 40]
lst[1:-1]	[20, 30, 40]
lst[:2]	[10, 30, 50]
lst[::-1]	[50, 40, 30, 20, 10]
and, or, not, True False	
<b>Import</b>	
from mod import mod1, mod2	mod1()
import mod	mod.mod1()
<b>Operators</b>	
	+ - // % *
<b>Operations</b>	
a	
.len()	
.sorted(list)	
list.sort()	
.reversed(list)	
list.reverse()	

Lists Operations	
.append()	[ , , []]
.extend()	[ , , , ]
.insert(index, value)	
.remove()	
<b>Dictionaries Operations</b>	
{key: value}	
d[key]	value
.update()	

Python (cont)	
.values()	
.keys()	
a	
<b>Strings</b>	
.count()	
.index()	
.upper()	
.capitalize()	
.split(" ")	
.strip(" ")	Remove todos os caracteres

MySQL	
SELECT column1, column2 FROM table_name WHERE x = "ads" ORDER BY column1 DESC	
SELECT COUNT(column_name) FROM table_name WHERE condition;	
SELECT AVG(column_name) FROM table_name WHERE condition;	
SELECT SUM(column_name) FROM table_name WHERE condition;	
SELECT column1, ... FROM table_name WHERE columnN LIKE pattern; % - any, _something	
INSERT INTO table name (column1, column2, column3, ...) VALUES (value1, value2, value3, ...)	
UPDATE table_name SET column1 = value1, column2 = value2, ... WHERE condition	
DELETE FROM table_name WHERE condition	



### MySQL (cont)

CREATE TABLE table\_name (column1 datatype, column2 datatype, ...)

DROP TABLE table\_name

DROP DATABASE databasename

ALTER TABLE table\_name ADD column\_name datatype;

ALTER TABLE table\_name DROP COLUMN column\_name;

ALTER TABLE table\_name MODIFY COLUMN column\_name datatype;

### Java

#### Arrays

.equals(Object[] arr1, Object[] arr2)

.sort(Object[] arr)

.toString()

#### String

.indexOf(char a)

.charAt(int index)

.equals(String s)

.format(String s, int 1, String 2, int 3, ...)

.length()

.substring(int beginIndex, int endIndex)

.toLowerCase()

.toUpperCase()

#### List<E>

.add(int index, E element)

.add (E element)

.contains(Object o)

.get(int index)

.indexOf(Object o)

### Java (cont)

.isEmpty()

.remove(int index)

.remove(Object o)

.set(int index, E Element)

.size()

.toArray()

.subList(int fromIndex, int toIndex)

HashMap<Object Key, Object Value>

.get(Object Key)

.put(Key key, Value value)

.remove(Object key)

.replace(K key, V value)

.size()

.values()

.containsKey(Object Key)

.containsValue(Object Value)

#### HashSet<E>

.add(E e)

.contains(Object o)

.remove(Object o)

.size()

.isEmpty()

.clear()

#### jUnits

void teste(){ ... } assertEquals(exp, comp)

### GIT

Criar repo git init

Aceder a staging area git status

Adicionar ficheiro a Staging Area git add {nome do ficheiro}

Remover da Staging Area git reset HEAD -- <file>

Dar commit git commit -m "- Commit message"

Criar novo branch git branch <new-branch-name>

Mudar de branch git checkout {branch}

Merge Branch git merge <branch-name>

Push para remoto git push <remote> <branch>

Atualizar git pull

Remover ficheiro do repositório git rm <file>

### Mongoose

Schema.create({ nome: name, password })

Schema.findOne({ \_id: id })

Schema.find({ name: name })

Schema.exists({ \_id: id })

Schema.updateMany({ \_id: id }, {name: name })

Schema.deleteOne({ \_id: id })



By Afonso Benedito

[cheatography.com/afonso-benedito/](https://cheatography.com/afonso-benedito/)

Published 19th June, 2022.

Last updated 20th June, 2022.

Page 2 of 3.

Sponsored by [CrosswordCheats.com](https://crosswordcheats.com)

Learn to solve cryptic crosswords!

<http://crosswordcheats.com>

### SOLID Principles

<b>Single Responsibility</b>	Class can have one responsibility
<b>Open/Closed</b>	Interfaces = - if's
<b>Liskov Substitution</b>	Subclass can do what Super does
<b>Interface Segregation</b>	Threedimensional shape x shape

### Dependency Inversion

### ExpressJS

```
const express = require('express');
```

```
Get app.get('/info', (req, res)=>{})
```

```
Post app.post('/:id', (req, res) => {})
```

```
Send res.status(200).send({})
```

```
Redirect res.redirect(301, '/')
```

### ReactJs

```
render(){return(<div>...</div>)}
```

```
Array => .map(() => { ... })
```

```
onClick = {() => this.handleSomething()}
```

```
Counters: <Counter Counter: this.prop-  
onDelete={}/> s.onDelete
```

### Class

```
state = {...json...}
```

```
this.setState({count: this.state.count + 1})
```

```
componentDidMount
```

```
componentDidUpdate
```

```
componentWillUnmount
```



By Afonso Benedito

[cheatography.com/afonso-benedito/](https://cheatography.com/afonso-benedito/)

Published 19th June, 2022.

Last updated 20th June, 2022.

Page 3 of 3.

Sponsored by [CrosswordCheats.com](https://CrosswordCheats.com)

Learn to solve cryptic crosswords!

<http://crosswordcheats.com>