

PROYECTO DE SOFTWARE

Cursada 2020

TEMARIO

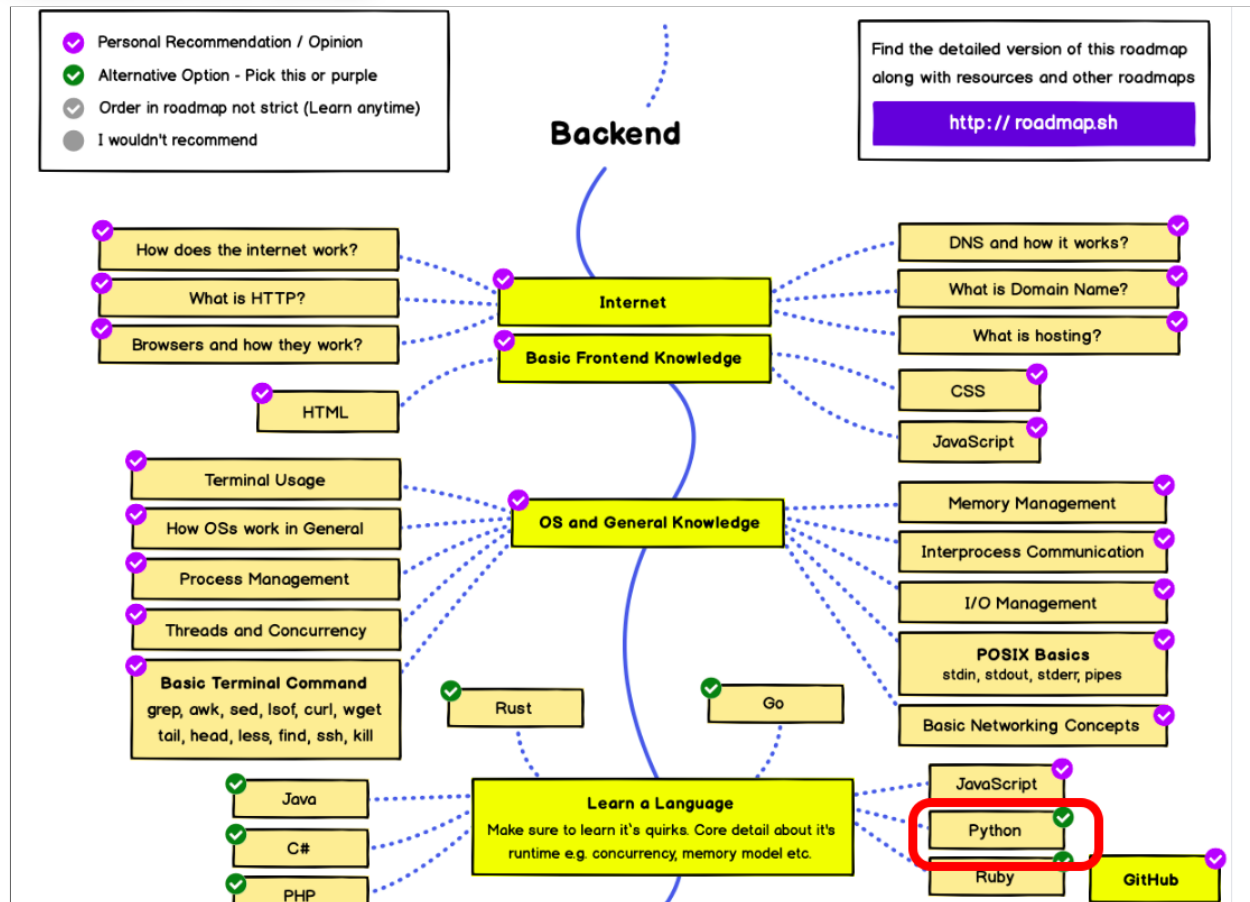
- Repaso Python.
- Entornos Virtuales.
- Flask

REPASO PYTHON

REPASANDO

- ¿Por qué usamos y promovemos el uso de software libre?
- ¿Características de Python?
- ¿Desarrollo en el servidor?

PROGRAMACIÓN EN EL SERVIDOR



NOSOTROS USAREMOS PYTHON

¿Por qué?

- <https://www.tiobe.com/tiobe-index/>
- <https://insights.stackoverflow.com/trends?tags=python>
- <https://github.info/>
- https://madnight.github.io/github/#/pull_requests/2019/2

PYTHON

- Fue desarrollado por **Guido Van Rossum** a finales de los años 80.
- El nombre proviene del grupo de cómicos ingleses **Monty Python**.
- Sitio Oficial: <http://python.org/>
- Python Argentina: <http://python.org.ar/>

CARACTERÍSTICAS GENERALES

- Es un lenguaje de alto nivel, fácil de aprender. Muy expresivo y legible.
- Es interpretado.
- Multiplataforma y multiparadigma.
- Posee una sintáxis elegante y tipado dinámico y fuerte.
- Tiene un manejo eficiente de estructuras de datos de alto nivel.

SINTAXIS BÁSICA

- **NO** hay declaración de variables: las variables se crean **dinámicamente** cuando se les asigna un valor.
- Tiene **tipado dinámico**: las variables pueden cambiar de tipo a lo largo del programa.
- Bloques, indentación.

ENTORNOS VIRTUALES

- Múltiples versiones de los lenguajes.
- Múltiples versiones de librerías.
- Requerimientos mínimos.

ENTORNOS VIRTUALES

- Herramienta que nos permite tener un ambiente para un proyecto con sus dependencias y librerías.
- Nos permite trabajar en un mismo equipo con diferentes versiones de lenguaje y paquetes.
- Virtualenv, Conda, ¿otros?

ENTORNOS VIRTUALES - PIP

- Pip es un sistema de gestión de paquetes.
- Primero, comprobamos si tenemos correctamente instalado pip.

```
sudo apt-get install python3-pip python3-dev build-essential
```

ENTORNOS VIRTUALES

- Instalamos virtualenv

```
pip3 install virtualenv
```

ENTORNOS VIRTUALES

- Nos posicionamos en la carpeta del proyecto o, en su defecto, la creamos:

```
mkdir miProyecto  
cd miProyecto
```

ENTORNOS VIRTUALES

- Inicializamos el entorno virtual:

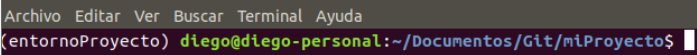
```
virtualenv entornoProyecto
```

```
(base) diego@diego-personal:~/Documentos/Git/miProyecto$ virtualenv entornoProyecto
Running virtualenv with interpreter /usr/bin/python2
New python executable in /home/diego/Documentos/Git/miProyecto/entornoProyecto/bin/python2
Also creating executable in /home/diego/Documentos/Git/miProyecto/entornoProyecto/bin/python
Installing setuptools, pkg_resources, pip, wheel...done.
```

ENTORNOS VIRTUALES

- Activamos el entorno virtual:

```
source entornoProyecto/bin/activate
```



```
Archivo Editar Ver Buscar Terminal Ayuda  
(entornoProyecto) diego@diego-personal:~/Documentos/Git/miProyecto$
```

- Para desactivar el entorno virtual:

```
deactivate
```


ENTORNOS VIRTUALES

- Nos facilita conocer cuáles son los paquetes y sus versiones para poder distribuirlo.

```
pip3 freeze > requirements.txt
```

- Podemos ver cuáles son los paquetes mirando requirements.txt

```
Click==7.0  
Flask==1.1.1  
itsdangerous==1.1.0  
Jinja2==2.10.1  
MarkupSafe==1.1.1  
pkg-resources==0.0.0  
Werkzeug==0.16.0
```

ENTORNOS VIRTUALES

- Para desplegar el proyecto en otra plataforma ejecutamos el siguiente comando:

```
pip3 install -r requirements.txt
```

ENTORNOS VIRTUALES

- Una herramienta útil para gestionar entornos virtuales es **virtualenvwrapper**:

```
pip3 install virtualenvwrapper
```

```
diego@diego-personal:~/Documentos/Git/miProyecto$ pip install virtualenvwrapper
Collecting virtualenvwrapper
  Downloading https://files.pythonhosted.org/packages/c1/6b/2f05d73b2d2f2410b48b90d3783a0034c26afa534a4a95ad5f1178d61191/virtualenvwrapper-4.8.4.tar.gz (334kB)
    100% |#####| 337kB 1.1MB/s
Collecting virtualenv (from virtualenvwrapper)
  Downloading https://files.pythonhosted.org/packages/8b/12/8d4f45b8962b03ac9efefe5ed5053f6b29334d83e438b4fe379d21c0cb8e/virtualenv-16.7.5-py2.py3-none-any.whl (3.3MB)
    100% |#####| 3.3MB 204kB/s
Collecting virtualenv-clone (from virtualenvwrapper)
  Downloading https://files.pythonhosted.org/packages/ba/f8/50c2b7dbc99e05fce5e5b9d9a31f37c988c99acd4e8dedd720b7b8d4011d/virtualenv_clone-0.5.3-py2.py3-none-any.whl
Collecting stevedore (from virtualenvwrapper)
  Downloading https://files.pythonhosted.org/packages/b1/e1/f5ddb83f60b03f522f173c03e406c1bff8343f0232a292ac96aa633b47a/stevedore-1.31.0-py2.py3-none-any.whl (43kB)
    100% |#####| 51kB 1.6MB/s
Collecting pbr!=2.1.0,>=2.0.0 (from stevedore->virtualenvwrapper)
  Downloading https://files.pythonhosted.org/packages/46/a4/d5c83831a3452713e4b4f126149bc4fbda170f7cb16a86a00ce57ce0e9ad/pbr-5.4.3-py2.py3-none-any.whl (110kB)
    100% |#####| 112kB 1.3MB/s
Collecting six>=1.10.0 (from stevedore->virtualenvwrapper)
  Cache entry deserialization failed, entry ignored
  Using cached https://files.pythonhosted.org/packages/73/fb/00a976f728d0d1fecfe898238ce23f502a721c0ac0ecfedb80e0d88c64e9/six-1.12.0-py2.py3-none-any.whl
Building wheels for collected packages: virtualenvwrapper
  Running setup.py bdist_wheel for virtualenvwrapper ... done
  Stored in directory: /home/diego/.cache/pip/wheels/70/d7/39/a522e494b0e145a1bec42f45a6e542f097c20d0be3ec26866e
Successfully built virtualenvwrapper
Installing collected packages: virtualenv, virtualenv-clone, pbr, six, stevedore, virtualenvwrapper
Successfully installed pbr-5.4.3 six-1.12.0 stevedore-1.31.0 virtualenv-16.7.5 virtualenv-clone-0.5.3 virtualenvwrapper-4.8.4
```

FLASK

- Instalamos un nuevo paquete al entorno, Flask.

```
(entornoProyecto) diego@diego-personal:~/Documentos/Git/miProyecto$ pip install flask
Collecting flask
  Downloading https://files.pythonhosted.org/packages/9b/93/628509b8d5dc749656a9641f4caf13540e2cdec85276964ff8f43bbb1d3b/Flask-1.1.1-py2.py3-none-any.whl (94kB)
    100% |#####| 102kB 857kB/s
Collecting click>=5.1 (from flask)
  Downloading https://files.pythonhosted.org/packages/fa/37/45185cb5abbc30d7257104c434fe0b07e5a195a6847506c074527aa599ec/Click-7.0-py2.py3-none-any.whl (81kB)
    100% |#####| 81kB 2.2MB/s
Collecting Werkzeug>=0.15 (from flask)
  Downloading https://files.pythonhosted.org/packages/ce/42/3aeda98f96e85fd26180534d36570e4d18108d62ae36f87694b476b83d6f/Werkzeug-0.16.0-py2.py3-none-any.whl (327kB)
    100% |#####| 327kB 934kB/s
Collecting itsdangerous>=0.24 (from flask)
  Downloading https://files.pythonhosted.org/packages/76/ae/44b03b253d6fade317f32c24d100b3b35c2239807046a4c953c7b89fa49e/itsdangerous-1.1.0-py2.py3-none-any.whl
Collecting Jinja2>=2.10.1 (from flask)
  Cache entry deserialization failed, entry ignored
  Downloading https://files.pythonhosted.org/packages/1d/e7/fd8b501e7a6dfe492a433deb7b9d833d39ca74916fa8bc63dd1a4947a671/Jinja2-2.10.1-py2.py3-none-any.whl (124kB)
    100% |#####| 133kB 1.6MB/s
Collecting MarkupSafe>=0.23 (from Jinja2>=2.10.1->flask)
  Cache entry deserialization failed, entry ignored
  Using cached https://files.pythonhosted.org/packages/fb/40/f3adb7cf24a8012813c5edb20329eb22d5d8e2a0ecf73d21d6b85865da11/MarkupSafe-1.1.1-cp27-cp27mu-manylinux1_x86_64.whl
Installing collected packages: click, Werkzeug, itsdangerous, MarkupSafe, Jinja2, flask
Successfully installed Jinja2-2.10.1 MarkupSafe-1.1.1 Werkzeug-0.16.0 click-7.0 flask-1.1.1 itsdangerous-1.1.0
(entornoProyecto) diego@diego-personal:~/Documentos/Git/miProyecto$
```

FLASK

- Hola Mundo

```
from flask import Flask
app = Flask(__name__)

@app.route('/')
def hello_world():
    return 'Hello, World!'
```

FLASK

- Miramos nuestro código en el browser:

```
export FLASK_APP=index.py
flask run
```

```
(entornoProyecto) diego@diego-personal:~/Documentos/Git/miProyecto$ flask run
* Serving Flask app "index.py"
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [22/Sep/2019 19:23:35] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [22/Sep/2019 19:23:35] "GET /favicon.ico HTTP/1.1" 404 -
```

FLASK

- Modificamos nuestro código:

```
from flask import Flask
app = Flask(__name__)

@app.route('/')
def hello_world():
    mensaje = 'Hello, World!'
    return mensaje + 2
```

Internal Server Error

The server encountered an internal error and was unable to complete your request. Either the server is overloaded or there is an error in the application.

FLASK

- Activamos el entorno de desarrollo:

```
export FLASK_ENV=development
```

```
^C(entornoProyecto) diego@diego-personal:~/Documentos/Git/miProyecto$ export FLASK_ENV=development
(entornoProyecto) diego@diego-personal:~/Documentos/Git/miProyecto$ flask run
* Serving Flask app "index.py" (lazy loading)
* Environment: development
* Debug mode: on
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger PIN: 177-792-228
```


FLASK

- Ahora podemos ver un poco más:

TypeError

TypeError: cannot concatenate 'str' and 'int' objects

Traceback (most recent call last)

```
File "/home/diego/Documentos/Git/miProyecto/entornoProyecto/lib/python2.7/site-packages/flask/app.py", line 2463, in __call__
    return self.wsgi_app(environ, start_response)
File "/home/diego/Documentos/Git/miProyecto/entornoProyecto/lib/python2.7/site-packages/flask/app.py", line 2449, in wsgi_app
    response = self.handle_exception(e)
File "/home/diego/Documentos/Git/miProyecto/entornoProyecto/lib/python2.7/site-packages/flask/app.py", line 1866, in handle_exception
    reraise(exc_type, exc_value, tb)
File "/home/diego/Documentos/Git/miProyecto/entornoProyecto/lib/python2.7/site-packages/flask/app.py", line 2446, in wsgi_app
    response = self.full_dispatch_request()
File "/home/diego/Documentos/Git/miProyecto/entornoProyecto/lib/python2.7/site-packages/flask/app.py", line 1951, in full_dispatch_request
    rv = self.handle_user_exception(e)
File "/home/diego/Documentos/Git/miProyecto/entornoProyecto/lib/python2.7/site-packages/flask/app.py", line 1820, in handle_user_exception
    reraise(exc_type, exc_value, tb)
File "/home/diego/Documentos/Git/miProyecto/entornoProyecto/lib/python2.7/site-packages/flask/app.py", line 1949, in full_dispatch_request
    rv = self.dispatch_request()
File "/home/diego/Documentos/Git/miProyecto/entornoProyecto/lib/python2.7/site-packages/flask/app.py", line 1935, in dispatch_request
    return self.view_functions[rule.endpoint](**req.view_args)
File "/home/diego/Documentos/Git/miProyecto/index.py", line 6, in hello_world
    return 'Hello, World!' + 2
```

TypeError: cannot concatenate 'str' and 'int' objects

FLASK

- Vamos recibir argumentos

```
from flask import Flask
app = Flask(__name__)

@app.route("/")
def index():
    return "Hola mundo"

@app.route("/personas")
def personas():
    return "personas"

@app.route("/personas/<string:nombre>")
def persona(nombre):
    nombre = nombre.capitalize()
    return f"Hola {nombre}"
```

FLASK

- Armamos un html:

```
from flask import Flask
app = Flask(__name__)

@app.route('/')
def hola_mundo():
    mensaje = '<!DOCTYPE html>'
    mensaje += '<html lang="es"> '
    mensaje += '<head>'
    mensaje += '</head>'
    mensaje += '<body>'
    mensaje += 'Hola mundo!'
    mensaje += '</body>'
    mensaje += '</html>'
    return mensaje
```

FLASK

- y otro...:

```
from flask import Flask
app = Flask(__name__)

@app.route("/personas")
def personas():
    mensaje = '<!DOCTYPE html>'
    mensaje += '<html lang="es"> '
    mensaje += '<head>'
    mensaje += '</head>'
    mensaje += '<body>'
    mensaje += 'Hola personas!'
    mensaje += '</body>'
    mensaje += '</html>'
    return mensaje
```

FLASK

- y otro...:

```
from flask import Flask
app = Flask(__name__)

@app.route("/personas/<string:nombre>")
def persona(nombre):
    nombre = nombre.capitalize()
    mensaje = '<!DOCTYPE html>'
    mensaje += '<html lang="es"> '
    mensaje += '<head>'
    mensaje += '</head>'
    mensaje += '<body>'
    mensaje += f"Hola {nombre}"
    mensaje += '</body>'
    mensaje += '</html>'
    return mensaje
```

FLASK

¿Problemas? ¿Inconvenientes?

FLASK

- Una solución

```
from flask import Flask, render_template

app = Flask(__name__)

@app.route("/")
def hola_mundo():
    return render_template ("index.html")
```

FLASK

- index.html

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Ejemplo Proyecto</title>
  </head>
  <body>
    <h1>Hola mundo!</h1>
  </body>
</html>
```


FLASK

- Con variables

```
@app.route("/personas")
def personas():
    contenido = "Hola personas!"
    return render_template("index1.html", contenido=contenido)

@app.route("/personas/<string:nombre>")
def persona(nombre):
    nombre = nombre.capitalize()
    contenido = f"Hola {nombre}"
    return render_template("index1.html", contenido=contenido)
```

FLASK

- index.html

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Ejemplo Proyecto</title>
  </head>
  <body>
    <h1>{{contenido}}</h1>
  </body>
</html>
```

FLASK

- Condicionales

```
@app.route("/")
def hola_mundo():
    return render_template("index.html", contenido="mundo")

@app.route("/personas")
def personas():
    contenido = "personas!"
    return render_template("index.html", contenido=contenido)

@app.route("/personas/<string:nombre>")
def persona(nombre):
    nombre = nombre.capitalize()
    return render_template("index.html", contenido=nombre)
```

FLASK

- index.html

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Ejemplo Proyecto</title>
  </head>
  <body>
    {% if contenido == "River" %}
      <h1>El campeón es {{contenido}}</h1>
    {% else %}
      <h1>Hola {{contenido}}</h1>
    {% endif %}
  </body>
</html>
```

FLASK

- Iteraciones

```
from flask import Flask, render_template

app = Flask(__name__)

pos_equipos = {"River" : "Campeón",
               "Boca": "Sub Campeón",
               "Gremio": "Semi finalista",
               "Palmeiras": "Semi finalista"}

@app.route("/equipos")
def personas():
    return render_template("index.html", contenido=pos_equipos)
```

FLASK

- index.html

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Ejemplo Proyecto</title>
  </head>
  <body>
    <ul>
      {% for equipo in contenido %}
        <li>{{equipo}}</li>
      {% endfor %}
    </ul>
  </body>
</html>
```

FLASK

- Url

```
app = Flask(__name__)

pos_equipos = {"River" : "Campeón",
               "Boca": "Sub Campeón",
               "Gremio": "Semi finalista",
               "Palmeiras": "Semi finalista"}

@app.route("/")
def index():
    return render_template ("index.html", contenido="mundo")

@app.route("/equipos")
def equipos():
    return render_template("equipos.html", contenido=pos_equipos)
```

FLASK

- index.html

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Ejemplo Proyecto</title>
  </head>
  <body>
    <h1> Libertadores 2018 </h1>
    <a href="{{url_for('equipos')}}"> Equipos </a>
  </body>
</html>
```


FLASK

- equipos.html

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Ejemplo Proyecto</title>
  </head>
  <body>
    <ul>
      {% for equipo in contenido %}
        <li>{{equipo}}</li>
      {% endfor %}
    </ul>
    <a href="{{url_for('index')}}"> Volver al inicio </a>
  </body>
</html>
```

FLASK

- Herencia
- layout.html

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Ejemplo Proyecto</title>
  </head>
  <body>
    <h1> {% block heading %} {% endblock %}</h1>
    {% block contenido %}
    {% endblock %}
  </body>
</html>
```

FLASK

- Herencia
- index.html

```
{% extends "layout.html"%}
{% block heading%}
    <h1> Libertadores 2018 </h1>
{% endblock %}
{% block contenido %}
    <a href="{{url_for('equipos')}}"> Equipos </a>
{% endblock %}
```

FLASK

- Herencia
- equipos.html

```
{% extends "layout.html"%}
{% block heading%}
    <h1> Posiciones 2018 </h1>
{% endblock %}
{% block contenido %}
    <ul>
        {% for equipo in contenido %}
            <li>{{equipo}}</li>
        {% endfor %}
    </ul>
    <a href="{{url_for('index')}}"> Volver al inicio </a>
{% endblock %}
```

FLASK

- Formularios

```
from flask import Flask, render_template, request

app = Flask(__name__)

pos_equipos = {"River" : "Campeón",
               "Boca": "Sub Campeón",
               "Gremio": "Semi finalista",
               "Palmeiras": "Semi finalista"}

@app.route("/equipos")
def equipos():
    return render_template("equipos.html", contenido=pos_equipos)

@app.route("/agregar", methods=["POST"])
def agregar():
    equipo = request.form.get("equipo")
    posicion = request.form.get("posicion")
    pos_equipos[equipo] = posicion
    return render_template("equipos.html", contenido=pos_equipos)
```

FLASK

- Formularios
- equipos.html

```
{% extends "layout.html"%}
{% block heading%}
    <h1> Posiciones 2018 </h1>
{% endblock %}
{% block contenido %}
    <ul>
        {% for equipo in contenido %}
            <li>{{equipo}}</li>
        {% endfor %}
    </ul>
    <form action="{{ url_for('agregar')}}" method="post">
        <input type="text" name="equipo" placeholder="Ingrese un equipo">
        <input type="text" name="posicion" placeholder="Ingrese la posicion">
        <button>Enviar</button>
    </form>
    <a href="{{url_for('index')}}"> Volver al inicio </a>
{% endblock %}
```

EJEMPLOS

<https://www.proyecto2020.linti.unlp.edu.ar/clase4/ejemplos/clase4.zip>

SEGUIMOS EN LA PRÓXIMA ...

<http://flask.palletsprojects.com/en/1.1.x/quickstart/>