

# KOLT Python

## Third-Party Packages

Ahmet Uysal

Monday 22<sup>nd</sup> April, 2019

# KOLT

1. Package Management  
○○○

2. Solving a Real Life Problem  
○○○○○○○○○

3. Final Project  
○

# Agenda

1. Package Management

2. Solving a Real Life Problem

3. Final Project



# Python Package Index (PyPI)

# Python Package Index (PyPI)

Repository of software for the Python programming language.

## Python Package Index (PyPI)

Repository of software for the Python programming language.

- 23,000+ Python3 packages.

## Python Package Index (PyPI)

Repository of software for the Python programming language.

- 23,000+ Python3 packages.
- If you want a package, PyPI probably has it.



## 1. Package Management



## 2. Solving a Real Life Problem



## 3. Final Project



# pip



# pip

- Recommended tool for installing Python packages.



# pip

- Recommended tool for installing Python packages.
- **pip** is already installed with modern Python distributions.

# pip

- Recommended tool for installing Python packages.
- **pip** is already installed with modern Python distributions.
- Try `pip -V` on your command line/terminal(`pip3 -V` for Macs).



# pip

- Recommended tool for installing Python packages.
- **pip** is already installed with modern Python distributions.
- Try `pip -V` on your command line/terminal(`pip3 -V` for Macs).

```
$ pip -V  
pip 19.0.3 from --PATH_TO_PIP-- (python 3.7)
```

# pip

- Recommended tool for installing Python packages.
- **pip** is already installed with modern Python distributions.
- Try `pip -V` on your command line/terminal(`pip3 -V` for Macs).

```
$ pip -V  
pip 19.0.3 from --PATH_TO_PIP-- (python 3.7)
```

## Any problems?

# Common pip commands

# Common pip commands

## Install a package:

## Common pip commands

### Install a package:

```
$ pip install package_name # latest version
```

# Common pip commands

## Install a package:

```
$ pip install package_name # latest version
```

```
$ pip install package_name==1.0.1 # specific version
```





# Common pip commands

## Install a package:

```
$ pip install package_name # latest version
```

```
$ pip install package_name==1.0.1 # specific version
```

```
$ pip install package_name>=1.0.1 # minimum version
```



## Common pip commands

### Install a package:

```
$ pip install package_name # latest version
```

```
$ pip install package_name==1.0.1 # specific version
```

```
$ pip install package_name>=1.0.1 # minimum version
```

### Uninstall a package:

## Common pip commands

### Install a package:

```
$ pip install package_name # latest version
```

```
$ pip install package_name==1.0.1 # specific version
```

```
$ pip install package_name>=1.0.1 # minimum version
```

### Uninstall a package:

```
$ pip uninstall package_name
```

# Common pip commands

## Install a package:

```
$ pip install package_name # latest version
```

```
$ pip install package_name==1.0.1 # specific version
```

```
$ pip install package_name>=1.0.1 # minimum version
```

## Uninstall a package:

```
$ pip uninstall package_name
```

## Update a package:

## Common pip commands

### Install a package:

```
$ pip install package_name # latest version
```

```
$ pip install package_name==1.0.1 # specific version
```

```
$ pip install package_name>=1.0.1 # minimum version
```

### Uninstall a package:

```
$ pip uninstall package_name
```

### Update a package:

```
$ pip install --upgrade package_name
```

## Common pip commands

### Install a package:

```
$ pip install package_name # latest version
```

```
$ pip install package_name==1.0.1 # specific version
```

```
$ pip install package_name>=1.0.1 # minimum version
```

### Uninstall a package:

```
$ pip uninstall package_name
```

### Update a package:

```
$ pip install --upgrade package_name
```

### Search PyPI for matches:

## Common pip commands

### Install a package:

```
$ pip install package_name # latest version
```

```
$ pip install package_name==1.0.1 # specific version
```

```
$ pip install package_name>=1.0.1 # minimum version
```

### Uninstall a package:

```
$ pip uninstall package_name
```

### Update a package:

```
$ pip install --upgrade package_name
```

### Search PyPI for matches:

```
$ pip search query
```

1. Package Management  
○○○

2. Solving a Real Life Problem  
●○○○○○○○

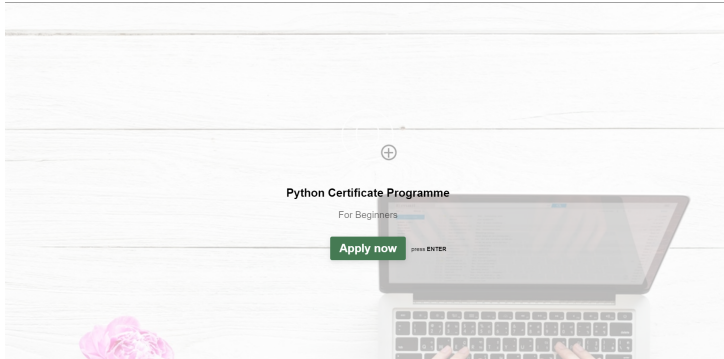
3. Final Project  
○

# Solving a Real Life Problem





# Solving a Real Life Problem



1. Package Management

○○○

2. Solving a Real Life Problem

●○○○○○○○

3. Final Project

○

# Problems



## Problems

- No option for Masters/PhD levels

## Problems

- No option for Masters/PhD levels
- No standard for study areas(majors)

## Problems

- No option for Masters/PhD levels
- No standard for study areas(majors)
- **What do you study in Koç?**

## Problems

- No option for Masters/PhD levels
- No standard for study areas(majors)
- **What do you study in Koç?**
  - Business Administration

## Problems

- No option for Masters/PhD levels
- No standard for study areas(majors)
- **What do you study in Koç?**
  - Business Administration
  - Bussines Administratiom

## Problems

- No option for Masters/PhD levels
- No standard for study areas(majors)
- **What do you study in Koç?**
  - Business Administration
  - Bussines Administratiom
  - Business



## Problems

- No option for Masters/PhD levels
- No standard for study areas(majors)
- **What do you study in Koç?**
  - Business Administration
  - Bussines Administratiom
  - Business
  - BA

## Problems

- No option for Masters/PhD levels
- No standard for study areas(majors)
- **What do you study in Koç?**
  - Business Administration
  - Bussines Administratiom
  - Business
  - BA
  - ...

## Problems

- No option for Masters/PhD levels
- No standard for study areas(majors)
- **What do you study in Koç?**
  - Business Administration
  - Bussines Administratiom
  - Business
  - BA
  - ...

Responses

260

Total visits

571

Unique visits

263

Completion rate

98.9%

Average time to complete

50:34



## How to Categorize Students?

We need academic level and major statistics of applicants.

## How to Categorize Students?

We need academic level and major statistics of applicants.  
What can we use?

## How to Categorize Students?

We need academic level and major statistics of applicants.  
What can we use?

- Jokes?

## How to Categorize Students?

We need academic level and major statistics of applicants.

What can we use?

- Jokes?
- Student mails?

## How to Categorize Students?

We need academic level and major statistics of applicants.

What can we use?

- Jokes?
- Student mails?
- KU usernames?





## How to Categorize Students?

We need academic level and major statistics of applicants.

What can we use?

- Jokes?
- Student mails?
- KU usernames?

`ldap.ku.edu.tr`



# How to Categorize Students?

The screenshot shows a web browser window with the address bar displaying "Not secure | ldap.ku.edu.tr/eGuide/servlet/eGuide". The page title is "eGuide™ Directory Search". Below the title bar, there are links for "Home", "Login", and "Help". The main content area is titled "Anonymous Search" and contains a search form with the following fields: "Find People", "Username" (with a dropdown arrow), "Equals" (with a dropdown arrow), and a text input field containing "AUYSAL16". To the right of the input field is a "Search" button with a magnifying glass icon and a "+" button. Below the search form, there is a table with the following columns: "First Name", "Last Name", "Department", and "Email". The table contains one row with the following data: "AHMET", "UYSAL", "Student CE ELEC", and "AUYSAL16@KU.E". To the right of the table, there is an "Information" tab and a "Print Format" link. Below the table, there is a section titled "Results 1 - 1 of 1".

First Name	Last Name	Department	Email
AHMET	UYSAL	Student CE ELEC	AUYSAL16@KU.E

Results 1 - 1 of 1

**Information**

**AHMET UYSAL** [Print Format](#)

Department: **Student CE ELEC**

Location:

Phone:

Email: [AUYSAL16@KU.EDU.TR](mailto:AUYSAL16@KU.EDU.TR)

Full Name : **AHMET UYSAL**

1. Package Management  
○○○

2. Solving a Real Life Problem  
○○○○●○○○

3. Final Project  
○

## Where did this information come?



## Where did this information come?

- Open the **developer console** on your browser.

## Where did this information come?

- Open the **developer console** on your browser.
- Windows: `Ctrl+Shift+I`, Mac: `Option+Command+I`

## Where did this information come?

- Open the **developer console** on your browser.
- Windows: `Ctrl+Shift+I`, Mac: `Option+Command+I`
- Go to **network** tab.

## Where did this information come?

- Open the **developer console** on your browser.
- Windows: `Ctrl+Shift+I`, Mac: `Option+Command+I`
- Go to **network** tab.
- Search for a person.

The screenshot shows a browser's developer network tab with the 'Preview' sub-tab selected. On the left, a list of resources is shown, with the request 'eGuide?User.context=oklzOaqoeNq&Action=Det...' selected. The main area displays the response content, which is a user profile for 'AHMET UYSAL'. The profile includes the following information:

- Department: Student CE ELEC
- Location:
- Phone:
- Email: [AUYSAL16@KU.EDU.TR](mailto:AUYSAL16@KU.EDU.TR)
- Full Name : AHMET UYSAL

There is also a 'Print Format' link and a 'powered by Novell eGuide' watermark.

# HTTP Requests



# HTTP Requests

Your browser sent a HTTP **GET** request to

`http://ldap.ku.edu.tr/eGuide/servlet/eGuide` to get the data.

# HTTP Requests

Your browser sent a HTTP **GET** request to

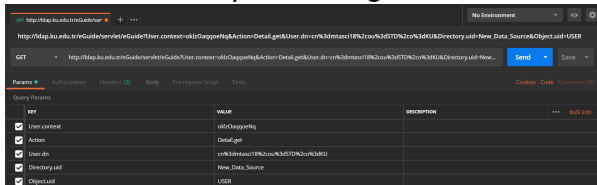
`http://ldap.ku.edu.tr/eGuide/servlet/eGuide` to get the data.

Examine the request using a tool like Postman.



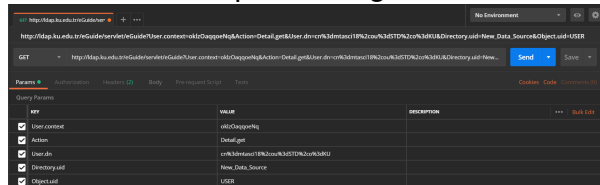
# HTTP Requests

Your browser sent a HTTP **GET** request to `http://ldap.ku.edu.tr/eGuide/servlet/eGuide` to get the data. Examine the request using a tool like Postman.



# HTTP Requests

Your browser sent a HTTP **GET** request to `http://ldap.ku.edu.tr/eGuide/servlet/eGuide` to get the data. Examine the request using a tool like Postman.



Can we send this request programmatically?

# requests package

`requests`: HTTP for Humans

## requests package

**requests**: HTTP for Humans

install using **pip**: `$ pip install requests`

## requests package

**requests**: HTTP for Humans

install using **pip**: `$ pip install requests`

*requests Quickstart*

## requests package

**requests**: HTTP for Humans

install using **pip**: `$ pip install requests`

*requests Quickstart*

```
import requests

response = requests.get('https://google.com')
print(response.content)
```





## import Statements

```
import foo                                # foo imported and bound locally
# below three statements imports foo.bar.baz
import foo.bar.baz                        # foo bound locally
import foo.bar.baz as fbb                 # foo.bar.baz bound as fbb
from foo.bar import baz                    # foo.bar.baz bound as baz
from foo import attr                       # foo imported, foo.attr bound as attr
```

## Problem Solution

You can get the source code [here](#).

## Problem Solution

You can get the source code [here](#).

Solution will be discussed during class time, you can ask your questions with email if you missed the class.

1. Package Management  
○○○

2. Solving a Real Life Problem  
○○○○○○○○○

3. Final Project  
●

# Final Project



1. Package Management  
○○○

2. Solving a Real Life Problem  
○○○○○○○○○

3. Final Project  
●

# Final Project

Final project is on!



# Final Project

Final project is on!

- You will work as teams of 2 or 3.

## Final Project

Final project is on!

- You will work as teams of 2 or 3.
- You can choose any topic, might be related to your research/other courses

## Final Project

Final project is on!

- You will work as teams of 2 or 3.
- You can choose any topic, might be related to your research/other courses
- Choose something you will enjoy working on



## Final Project

Final project is on!

- You will work as teams of 2 or 3.
- You can choose any topic, might be related to your research/other courses
- Choose something you will enjoy working on
- We will spend most of our remaining class time working on your projects

## Final Project

Final project is on!

- You will work as teams of 2 or 3.
- You can choose any topic, might be related to your research/other courses
- Choose something you will enjoy working on
- We will spend most of our remaining class time working on your projects
- Proposals due next Monday!