First Project in postgresql

Orange is all about data visualizations that help to uncover hidden data patterns, provide intuition behind data analysis procedures or support communication between data scientists and domain experts. Visualization widgets include scatter plot, box plot and histogram, and model-specific visualizations like dendrogram, silhouette plot, and tree visualizations, just to mention a few. Many other visualizations are available in add-ons and include visualizations of networks, word clouds, geographical maps, and more.  
  
  
  
We take care to make Orange visualizations interactive: you can select data points from a scatter plot, a node in the tree, a branch in the dendrogram. Any such interaction will instruct visualization to send out a data subset that corresponds to the selected part of visualization. Consider the combination of a scatter plot and classification tree below. Scatter plot shows all the data, but highlights the data subset that corresponds to the selected node in the classification tree.  
  
  
  
  
  
  
  
Great Visualizations  
  
  
  
Orange includes many standard visualizations. Scatter plot is great for visualizing correlations between pair of attributes, box plot for displaying basic statistics, heat map to provide an overview across entire data set, and projection plots like MDS for plotting the multinomial data in two dimensions.  
  
  
  
  
  
  
  
Besides visualizations one would expect in a data mining suite, Orange includes some great extras that you may not find in other packages. These include widgets for silhouette plot to analyze the results of clustering, mosaic and Sieve diagram to discover feature interactions, and Pythagorean tree visualization for classification trees and forests.  
  
  
  
  
  
  
  
Exploratory Data Analysis  
  
  
  
Interactive visualizations enable exploratory data analysis. One can select interesting data subsets directly from plots, graphs and data tables and mine them in them downstream widgets. For example, select a cluster from the dendrogram of hierarchical clustering and map it to a 2D data presentation in the MDS plot. Or check their values of in the data table. Or observe the spread of its feature values in a box plot. Open all these windows at once and see how the changes in your selection affect other widgets. Or, for another example, cross-validate logistic regression on a data set and map some of the misclassifications to the two-dimensional projection. It is easy to turn Orange into a tool where domain experts can explore their data even if they lack insights in underlying statistics or machine learning.

## First Project in postgresql` Viewpoints

### 1: Programmer`s role to the society

This will create a dumpfile of the data stored in your SQLite database. the--natural-foreignargument serializes foreign keys, since you are transitioning to a new database. The-e contenttypes -e auth.Permissionarguments exclude tables that would cause Django to throw anIntegrityError. Then, install the relevent dependencies in your Python environment usingpiporconda:  
  
  
  
  
  
$ pip install MySQL-python  
  
  
  
  
  
If you are using a virtual environment for Python, make sure you havepipinstalled in your virtual environment before using thepipcommands above, or else the dependencies will be installed globally.  
  
  
  
MySQL-pythonis a self-contained driver that enables Python programs to interface with MySQL databases.mysql-connector-pythonis another database connector option, created and maintained by Oracle. The difference between the two is thatmysql-connector-pythonis written in Python whileMySQL-pythonis written in C. I suggest to useMySQL-pythonbecause it is faster with almost all SQL commands. The downside is that it is not compliant with Python 3.  
  
  
  
Finally, in yoursettings.pyfile in your app, change theDATABASESsection to the following:

## First Project in postgresql` Goals

### 1: Programmers Practice

A Goal from Programmer`s role to the society

This will create a dumpfile of the data stored in your SQLite database. the--natural-foreignargument serializes foreign keys, since you are transitioning to a new database. The-e contenttypes -e auth.Permissionarguments exclude tables that would cause Django to throw anIntegrityError. Then, install the relevent dependencies in your Python environment usingpiporconda:  
  
  
  
  
  
$ pip install MySQL-python  
  
  
  
  
  
If you are using a virtual environment for Python, make sure you havepipinstalled in your virtual environment before using thepipcommands above, or else the dependencies will be installed globally.  
  
  
  
MySQL-pythonis a self-contained driver that enables Python programs to interface with MySQL databases.mysql-connector-pythonis another database connector option, created and maintained by Oracle. The difference between the two is thatmysql-connector-pythonis written in Python whileMySQL-pythonis written in C. I suggest to useMySQL-pythonbecause it is faster with almost all SQL commands. The downside is that it is not compliant with Python 3.  
  
  
  
Finally, in yoursettings.pyfile in your app, change theDATABASESsection to the following:

### 2: Hello world

A Goal from Programmer`s role to the society

This will create a dumpfile of the data stored in your SQLite database. the--natural-foreignargument serializes foreign keys, since you are transitioning to a new database. The-e contenttypes -e auth.Permissionarguments exclude tables that would cause Django to throw anIntegrityError. Then, install the relevent dependencies in your Python environment usingpiporconda:  
  
  
  
  
  
$ pip install MySQL-python  
  
  
  
  
  
If you are using a virtual environment for Python, make sure you havepipinstalled in your virtual environment before using thepipcommands above, or else the dependencies will be installed globally.  
  
  
  
MySQL-pythonis a self-contained driver that enables Python programs to interface with MySQL databases.mysql-connector-pythonis another database connector option, created and maintained by Oracle. The difference between the two is thatmysql-connector-pythonis written in Python whileMySQL-pythonis written in C. I suggest to useMySQL-pythonbecause it is faster with almost all SQL commands. The downside is that it is not compliant with Python 3.  
  
  
  
Finally, in yoursettings.pyfile in your app, change theDATABASESsection to the following:

### 3: Programmers Gamming

A Goal from Programmer`s role to the society

This will create a dumpfile of the data stored in your SQLite database. the--natural-foreignargument serializes foreign keys, since you are transitioning to a new database. The-e contenttypes -e auth.Permissionarguments exclude tables that would cause Django to throw anIntegrityError. Then, install the relevent dependencies in your Python environment usingpiporconda:  
  
  
  
  
  
$ pip install MySQL-python  
  
  
  
  
  
If you are using a virtual environment for Python, make sure you havepipinstalled in your virtual environment before using thepipcommands above, or else the dependencies will be installed globally.  
  
  
  
MySQL-pythonis a self-contained driver that enables Python programs to interface with MySQL databases.mysql-connector-pythonis another database connector option, created and maintained by Oracle. The difference between the two is thatmysql-connector-pythonis written in Python whileMySQL-pythonis written in C. I suggest to useMySQL-pythonbecause it is faster with almost all SQL commands. The downside is that it is not compliant with Python 3.  
  
  
  
Finally, in yoursettings.pyfile in your app, change theDATABASESsection to the following:

### 4: Programmers Gamming

A Goal from Programmer`s role to the society

This will create a dumpfile of the data stored in your SQLite database. the--natural-foreignargument serializes foreign keys, since you are transitioning to a new database. The-e contenttypes -e auth.Permissionarguments exclude tables that would cause Django to throw anIntegrityError. Then, install the relevent dependencies in your Python environment usingpiporconda:  
  
  
  
  
  
$ pip install MySQL-python  
  
  
  
  
  
If you are using a virtual environment for Python, make sure you havepipinstalled in your virtual environment before using thepipcommands above, or else the dependencies will be installed globally.  
  
  
  
MySQL-pythonis a self-contained driver that enables Python programs to interface with MySQL databases.mysql-connector-pythonis another database connector option, created and maintained by Oracle. The difference between the two is thatmysql-connector-pythonis written in Python whileMySQL-pythonis written in C. I suggest to useMySQL-pythonbecause it is faster with almost all SQL commands. The downside is that it is not compliant with Python 3.  
  
  
  
Finally, in yoursettings.pyfile in your app, change theDATABASESsection to the following:

## First Project in postgresql` Requirements

### 1: The first Requirement

This will create a dumpfile of the data stored in your SQLite database. the--natural-foreignargument serializes foreign keys, since you are transitioning to a new database. The-e contenttypes -e auth.Permissionarguments exclude tables that would cause Django to throw anIntegrityError. Then, install the relevent dependencies in your Python environment usingpiporconda:  
  
  
  
  
  
$ pip install MySQL-python  
  
  
  
  
  
If you are using a virtual environment for Python, make sure you havepipinstalled in your virtual environment before using thepipcommands above, or else the dependencies will be installed globally.  
  
  
  
MySQL-pythonis a self-contained driver that enables Python programs to interface with MySQL databases.mysql-connector-pythonis another database connector option, created and maintained by Oracle. The difference between the two is thatmysql-connector-pythonis written in Python whileMySQL-pythonis written in C. I suggest to useMySQL-pythonbecause it is faster with almost all SQL commands. The downside is that it is not compliant with Python 3.  
  
  
  
Finally, in yoursettings.pyfile in your app, change theDATABASESsection to the following:

## The first Requirement` Scenarios

### 1: new one

jkhjkhjkhjkhfdjkfj dfkjfdkj

## The first Requirement` Processes

### 1: The first created Process 2

jkhkhjkhjkhvjkhjkhcvjkhvjkchvjkch

### 2: The second Requirement

This will create a dumpfile of the data stored in your SQLite database. the--natural-foreignargument serializes foreign keys, since you are transitioning to a new database. The-e contenttypes -e auth.Permissionarguments exclude tables that would cause Django to throw anIntegrityError. Then, install the relevent dependencies in your Python environment usingpiporconda:  
  
  
  
  
  
$ pip install MySQL-python  
  
  
  
  
  
If you are using a virtual environment for Python, make sure you havepipinstalled in your virtual environment before using thepipcommands above, or else the dependencies will be installed globally.  
  
  
  
MySQL-pythonis a self-contained driver that enables Python programs to interface with MySQL databases.mysql-connector-pythonis another database connector option, created and maintained by Oracle. The difference between the two is thatmysql-connector-pythonis written in Python whileMySQL-pythonis written in C. I suggest to useMySQL-pythonbecause it is faster with almost all SQL commands. The downside is that it is not compliant with Python 3.  
  
  
  
Finally, in yoursettings.pyfile in your app, change theDATABASESsection to the following:

## The second Requirement` Scenarios

### 1: new one

jkhjkhjkhjkhfdjkfj dfkjfdkj

## The second Requirement` Processes

### 1: The first created Process 2

jkhkhjkhjkhvjkhjkhcvjkhvjkchvjkch

### 3: water

This will create a dumpfile of the data stored in your SQLite database. the--natural-foreignargument serializes foreign keys, since you are transitioning to a new database. The-e contenttypes -e auth.Permissionarguments exclude tables that would cause Django to throw anIntegrityError. Then, install the relevent dependencies in your Python environment usingpiporconda:  
  
  
  
  
  
$ pip install MySQL-python  
  
  
  
  
  
If you are using a virtual environment for Python, make sure you havepipinstalled in your virtual environment before using thepipcommands above, or else the dependencies will be installed globally.  
  
  
  
MySQL-pythonis a self-contained driver that enables Python programs to interface with MySQL databases.mysql-connector-pythonis another database connector option, created and maintained by Oracle. The difference between the two is thatmysql-connector-pythonis written in Python whileMySQL-pythonis written in C. I suggest to useMySQL-pythonbecause it is faster with almost all SQL commands. The downside is that it is not compliant with Python 3.  
  
  
  
Finally, in yoursettings.pyfile in your app, change theDATABASESsection to the following:

## water` Scenarios

### 1: Second Scenario to be created

goal.

### 2: one

utu

### 3: new one

jkhjkhjkhjkhfdjkfj dfkjfdkj

## water` Processes

### 1: process

process

### 2: The first created Process 2

jkhkhjkhjkhvjkhjkhcvjkhvjkchvjkch