My Project

Generated by Doxygen 1.9.5

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

Namespace Documentation

3.1 Multiple Namespace Reference

Client Server Module Using Multithreading This module consist of Server Code that can handle multiple customers(clients) at the same time for the Shopping Mall based Model.

3.1.1 Detailed Description

Client Server Module Using Multithreading This module consist of Server Code that can handle multiple customers(clients) at the same time for the Shopping Mall based Model.

The Customers will be provided with the Interface with which they can select multiple operations and also choose the shop they want to go and by using the current data if any (or will be created) they will be provided with suggestion regarding the next shop they should visit. More details.

Chapter 4

Class Documentation

4.1 project01.user Class Reference

Class named User consist of all the Methods that are required for the handling of Clients and all the data calculations required for the suggestions that will be provided to the customer.

Public Member Functions

- def **__init__** (self, username)

  The constructor.
- def run (self)

  Run Method handle or provide with the Operation List that can be performed in order to choose the shop customer would like to go next or taking suggestions and also see their current path of shops they visited.
- def create_data (self)

  Create Data generates random data that will be further useful for the suggestion.
- def new_user (self, user_name)

  New User creates the New User and Save its Name in New User DataBase.
- def append_list (self)

  Append List let User to choose the next shop that user want to visit and than once User has visited all the shop once it stores it into the DataBase it also let user know that same shop donot get repeated.
- def suggestion (self)

  Suggestion uses the current user data and compares it within the DataBase and it return suggestion list on the basis of patter matching.
- def row_count (self)

  Row Count returns the total number of users present in the DataBase.
- def data_match (self, customer_data)

  Data Match does pattern matching with the exisiting Data Base in order to find the next shop user should visit.
- def print_list (self)

  Print_list prints the current list of the shops that user has visited.
- def client_handler (self, connection)

  client_handler informs the client that is now connected and furhter start the run process.
- def accept_connections (self, ServerSocket)

  accept_connection accepts the connection that is requested by every new client.
- def start_server (self, host, port)

  start_server bind the port on which the communication between server and client will take place.
- def send_data (self, message)

  send_data will send the data on the connected client.
- def receive_data (self)

  receive_data receives the data that is being send by client.

Public Attributes

- **name**

4.1.1 Detailed Description

Class named User consist of all the Methods that are required for the handling of Clients and all the data calculations required for the suggestions that will be provided to the customer.

More details.

4.1.2 Member Function Documentation

4.1.2.1 accept_connections()

```
def project01.user.accept_connections (
              self,
              ServerSocket )
```

accept_connection accepts the connection that is requested by every new client.

Parameters

self	The object pointer.
ServerSocket	The Socket on which request will be initiated

4.1.2.2 append_list()

```
def project01.user.append_list (
              self )
```

Append List let User to choose the next shop that user want to visit and than once User has visited all the shop once it stores it into the DataBase it also let user know that same shop donot get repeated.

Parameters

self	The object pointer.

4.1.2.3 client_handler()

```
def project01.user.client_handler (
              self,
              connection )
```

client_handler informs the client that is now connected and furhter start the run process.

Parameters

self	The object pointer.
connection	The Socket Connection

4.1.2.4 create_data()

```
def project01.user.create_data (
              self )
```

Create Data generates random data that will be further useful for the suggestion.

Parameters

self	The object pointer.

4.1.2.5 data_match()

```
def project01.user.data_match (
              self,
              customer_data )
```

Data Match does pattern matching with the exisiting Data Base in order to find the next shop user should visit.

Parameters

self	The object pointer.
customer_data	The List of the Exisiting Customer Data

4.1.2.6 new_user()

```
def project01.user.new_user (
              self,
              user_name )
```

New User creates the New User and Save its Name in New User DataBase.

Parameters

self	The object pointer.
user_name	The Name Entered by User

4.1.2.7 print_list()

```
def project01.user.print_list (
              self )
```

Print_list prints the current list of the shops that user has visited.

Parameters

self	The object pointer.

4.1.2.8 receive_data()

```
def project01.user.receive_data (
              self )
```

receive_data receives the data that is being send by client.

Parameters

self	The object pointer.

4.1.2.9 row_count()

```
def project01.user.row_count (
              self )
```

Row Count returns the total number of users present in the DataBase.

Parameters

self	The object pointer.

4.1.2.10 run()

```
def project01.user.run (
              self )
```

Run Method handle or provide with the Operation List that can be performed in order to choose the shop customer would like to go next or taking suggestions and also see their current path of shops they visited.

Parameters

self	The object pointer.

4.1.2.11 send_data()

```
def project01.user.send_data (
              self,
              message )
```

send_data will send the data on the connected client.

Parameters

self	The object pointer.
message	The data to be sent to client.

4.1.2.12 start_server()

```
def project01.user.start_server (
              self,
              host,
              port )
```

start_server bind the port on which the communication between server and client will take place.

Parameters

self	The object pointer.
host	The Host ID.
port	The Port on which communication will take place.

4.1.2.13 suggestion()

```
def project01.user.suggestion (
              self )
```

Suggestion uses the current user data and compares it within the DataBase and it return suggestion list on the basis of patter matching.

Parameters

self	The object pointer.

The documentation for this class was generated from the following file:

- project01.py

Index