**Documentation of the python program (mail.py)**

“mail.py” is a RPA program to send automated Emails.

Here are the roles of the code line by line given below\_

|  |  |
| --- | --- |
| **Line no. of code** | **Role of code** |
| 1. | Import logging library. |
| 2. | Import OS module. |
| 3. | Import library to send any attachment. |
| 4. | Import library to send any messages. |
| 5. | Import library to send email to any internet machine. |
| 6. | Import library to encode the attached files. |
| 7. | Import library to get access of attachments. |
| 8. | Import library for data analysis. |
| 9. | Import library to create tables into the mail body. |
| 10. | Import ‘Config’ class from “utils.config” file. |
| 11. | Import ‘Logger’ class from “utils.logger” file. |
| 16. | Creating the class “Mail”. |
| 23. | Get the instance variable of Logger and copy into ‘logger’ variable. |
| 24. | Copy the file name into the variable. |
| 27-39 | Declaring the variables. |
| 40-44 | Set variable with plain text. |
| 46-59 | Set variable with html script. I will also create a message. |
| 61-71 | Whenever this function will be called. It will import the user’s server, port, username, password and email from which the mail will be sent. |
| 73-112 | Whenever this function will compose the mail. |
| 87. | It will show email address where the mail will be sent. |
| 89-98 | For creating table to compose the message for the mail.  It will insert the html text (of line 46-59) and plain text (of line 40-44) into the table. |
| 100-101 | Will make a copy of the receivers email address into the variables. |
| 103-104 | It will attach the text tables into the mail. |
| 106. | It will attach the mail subject into the mail. |
| 107. | It will attach the receiver’s email address into the mail. |
| 108. | It will attach the sender’s email address into the mail. |
| 109-110 | It will attach the Carbon Copy email addresses into the mail. |
| 111-112 | It will attach the attachment files if there is any. |
| 114-123 | This function will send the email. |
| 119. | Will show the message “Sending Email” while executing the operation. |
| 120. | It will set the server and the port using ‘smptlib’ library. |
| 121. | Log in with the user’s email. |
| 122. | Will send the email. |
| 123. | Exit from the server. |
| 125-161 | It will attach the attachment files into the mail if there is any. |
| 133. | Will check if the file type is ‘String on not’. |
| 134. | If it is ‘true’, then it will be opened as ‘attachment’. |
| 135. | It will set the file’s maintype as ‘application’ and subtype as ‘octet-stream’. And it will allow python to upload the file. |
| 136. | It will read the attachment file. |
| 138. | It will encode the attachment as base64, which is very suitable for sending emails. |
| 139. | It will copy the attachment name into the variable. |
| 141. | It will add the header as per file’s name. |
| 142. | It will finally attach the attachment file into the email. |
| 146-159 | It will repeat the process of 133-142th line, if the attachment type is ‘list’. |
| 160-161 | It will display an error message if the type of the attachment is neither ‘String’ nor ‘list of string’. |
|  |  |
|  |  |
|  |  |
|  |  |