GENERAL OUTLINE OF THE PROJECT:

- Structure.
- Function (file containing available currencies relation with dollar).
- Function (to remove the above presented file).
- Function (to get structure pointer value).
- 2 Function (to get strings in uppercase letters).
- Function (to get currencies).
- 120+ Function various for conversion.
- All the currency data are stored in the file.

PROJECT BRIEF DESCRIPTION:

- Firstly, a global structure currency is initialized which contains one **float value** (amount) and two **strings** with names of **To**(the currency to be converted into) and **from**(the currency that you have)and a structure p1 is initialized globally.
- The project reads a file called **currency.txt** which contains the relation of all the available currencies with the US dollar and displays it. the function used to do this is **void dollar_conversion_file(void);.**
- if the file is unable to open or the file does not exist the program will exit.
- Another function remove() is used which asks the user if he wants to keep the conversions displayed or if he does not require it and wants to remove it .
- Now another file is read and it displays all the available currencies that can be converted.
- Now a function **void getpoint(currency &p1)** is used to get inputs in the structure p1. this function prompts the user to enter the amount and the names of currencies.
- After this another function is used which converts the names of currencies entered by the user to uppercase letters so that it works for both uppercase and lowercase and mixed letters.

NOTE: **have use algorithm [`transform(p1.To.begin(), p1.To.end(), p1.To.begin(), ::toupper);here p1.to and p1.from is used at the place of string(as named in function).`**

- **Now another** function float **givencurrency(currency&)** is defined. The function **getpoint(currency p1)** is called in this function. The function call is **givencurrency(p1)** which is used in the main().
- This function takes p1 as argument and checks the names of the currencies provided by user by comparing it with other currencies using if and if else conditions.
- If the information entered by the user is satisfied by a condition it converts the amount entered by user by proper calculation which is done by a function.
- Every possible conversion has its own function whose return type is float and returns the calculated amount which calculates the converted amount e.g

```
 else if (((p1.from == "INDONASIA_RUPIAH") && (p1.To == "AFGHAN_AFGHANI"))
{
   cout << "YOUR ANSWER IS :";
   //1 indonesian rupiah = 0.0055 afghan afghani
   answer = givenresult2(p1);
   cout << answer;

}
```

prototype:

```
float givenresult2(currency& p1)
{
   float x = p1.value * 0.0055;
   return x;
}
```

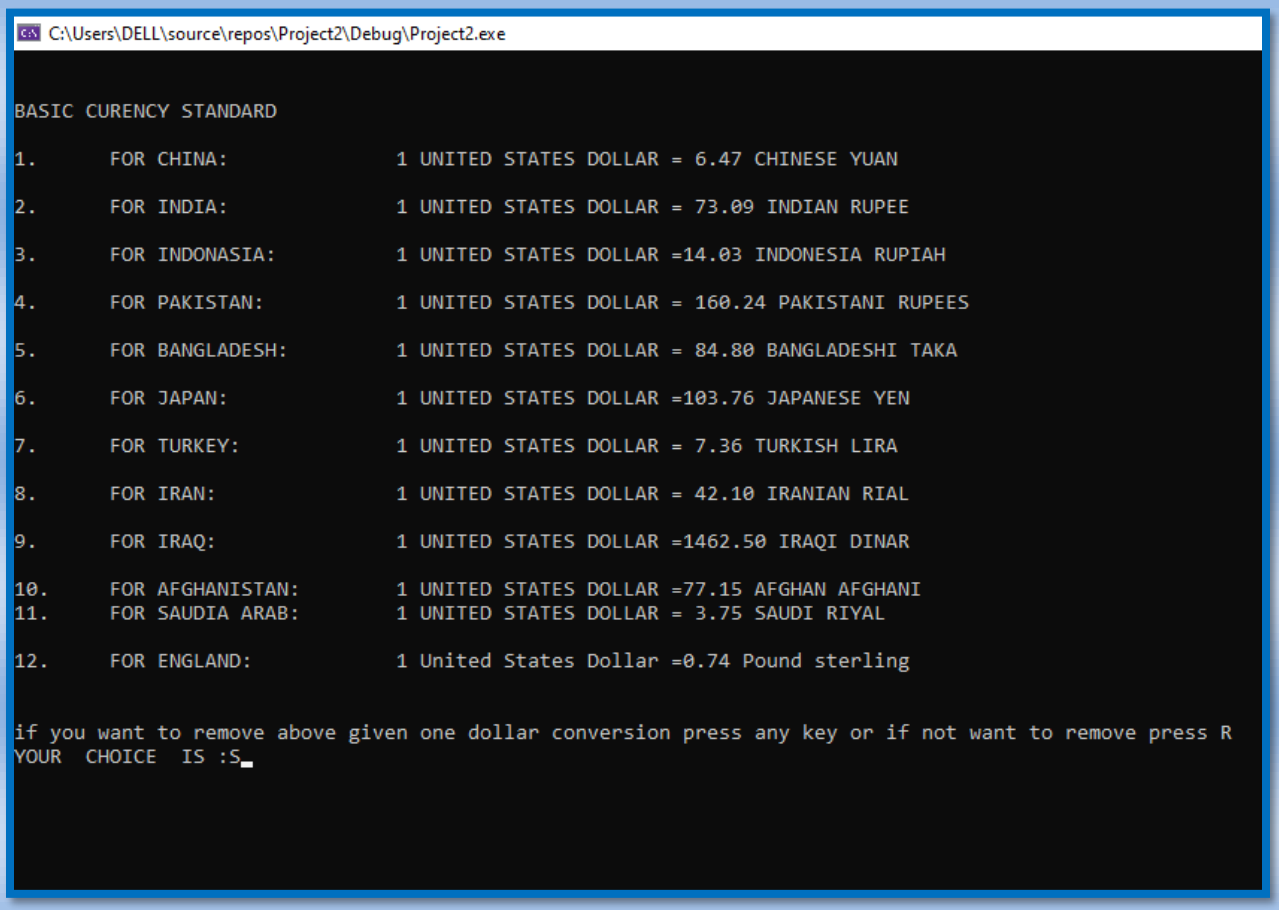
- And if the information entered by the user does not satisfy any condition then the prograam says that either the currency is not available or you have entered wrong spellings and then asks the user if he wants to input again in case he entered wrong spellings or if he wants to exit.
- The converted amount is stored in a variable called answer which is of type float.
- this function then returns the value of answer as its return type is also float.
- this function is called in the main() and its float value is stored in float result i.e float result= givencurrency(p1).

- Now a function is used to store the record in a file void storecontentinfile(currency&, float);. This function is called in the main() as storecontentinfile(p1, result); this stores the original amount the names of both currencies and the converted amount.
- after this, the program askes the user if he wants to use the program again or if he wants to exit. switch statements and labels were used and if he continues the next recored will save in file. so the user can use this program as many times as he wants.

"PROJECT OUTPUTS:"

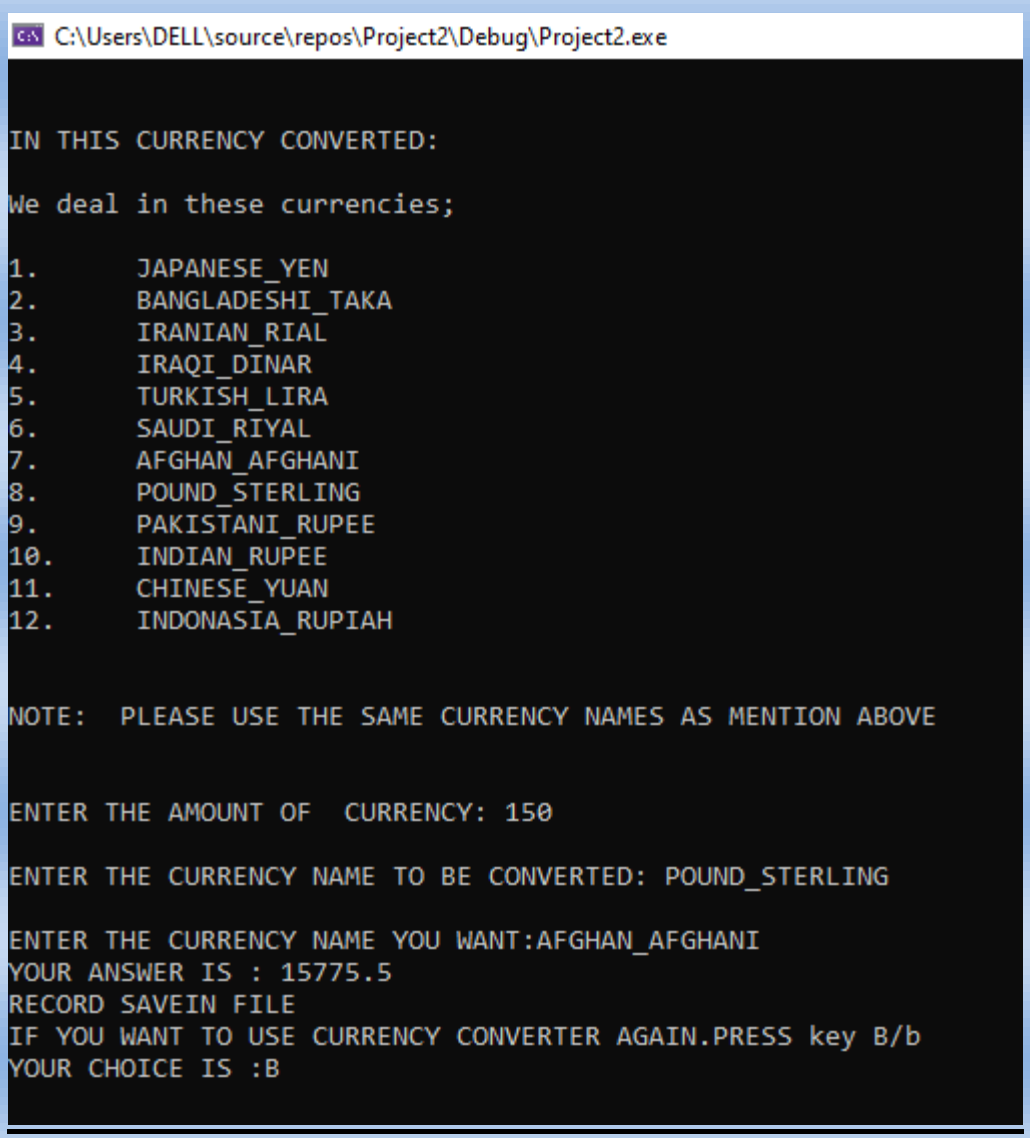
Output showing currency used rate with dollar.

OUTPUT#1:



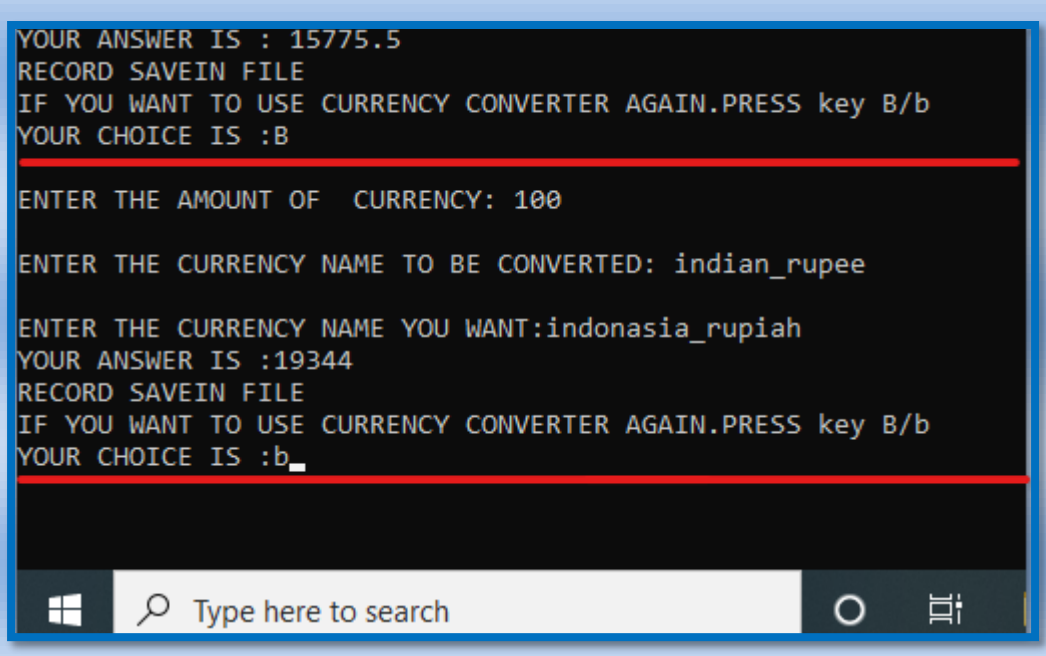
OUTPUT#2:

Currencies are entered in upper case.



OUTPUT#3:

Currencies are used in small case but it gives the output.



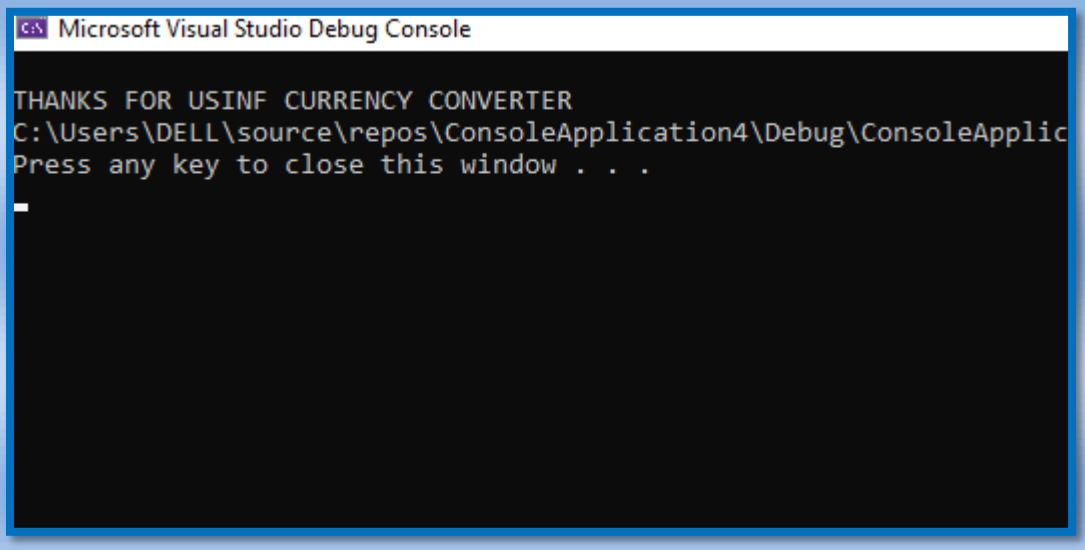
OUTPUT#4

Currencies are used in small mixed case letters but it gives the output.

```
RECORD SAVEIN FILE
IF YOU WANT TO USE CURRENCY CONVERTER AGAIN.PRESS key B/b
YOUR CHOICE IS :b

ENTER THE AMOUNT OF  CURRENCY: 100

ENTER THE CURRENCY NAME TO BE CONVERTED: Chinese_yuan

ENTER THE CURRENCY NAME YOU WANT:pakistani_RUPEE
OUR ANSWER IS : 2486
RECORD SAVEIN FILE
IF YOU WANT TO USE CURRENCY CONVERTER AGAIN.PRESS key B/b
YOUR CHOICE IS :f
```

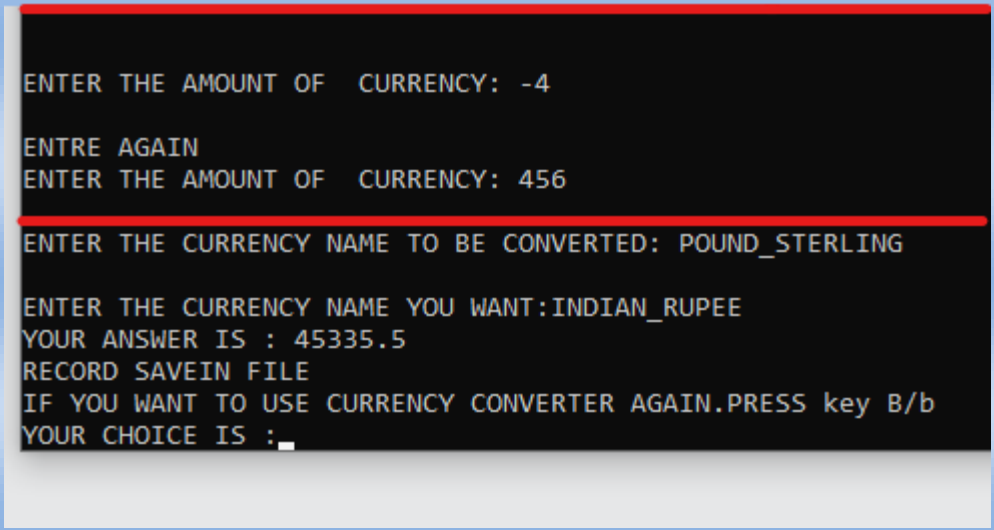
OUTPUT #5:

The output clearly shows that all the above data is cleared from the screen and it shows a message as an output and exist the program.



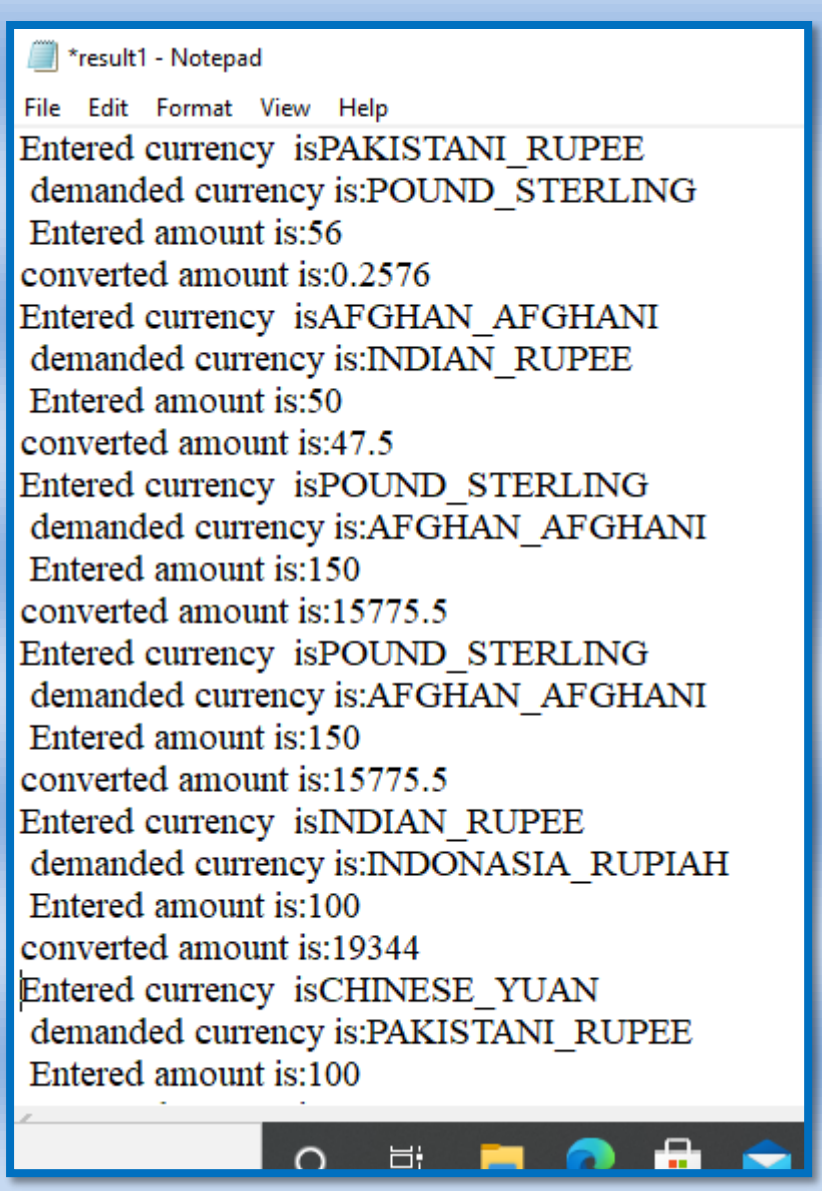
OUTPUT#6:

If the user enter currency in -1, it will ask the user to enter again.



OUTPUT #7:

File showing all save results:


*result1 - Notepad
File Edit Format View Help
Entered currency isPAKISTANI_RUPEE
demanded currency is:POUND_STERLING
Entered amount is:56
converted amount is:0.2576
Entered currency isAFGHAN_AFGHANI
demanded currency is:INDIAN_RUPEE
Entered amount is:50
converted amount is:47.5
Entered currency isPOUND_STERLING
demanded currency is:AFGHAN_AFGHANI
Entered amount is:150
converted amount is:15775.5
Entered currency isPOUND_STERLING
demanded currency is:AFGHAN_AFGHANI
Entered amount is:150
converted amount is:15775.5
Entered currency isINDIAN_RUPEE
demanded currency is:INDONASIA_RUPIAH
Entered amount is:100
converted amount is:19344
Entered currency isCHINESE_YUAN
demanded currency is:PAKISTANI_RUPEE
Entered amount is:100

THE END!!