**University Of Gujrat**

**Name: Mariam Khalid**

**Roll no:18321519-085**

**Class:BS-CS-18(B)**

**Submitted to :DR.Nouman Riaz**

**Code:**

#include <iostream>

#include <fstream>

using namespace std;

class HOD

{

public:

// Attributes

string name, surename, gender;

int age;

public:

// SETTERS

void set\_name(string n)

{

name = n;

}

void set\_surname(string s)

{

surename = s;

}

void set\_gender(string g)

{

gender = g;

}

void set\_age(int a)

{

age = a;

}

//GETTERS

string get\_name()

{

return this->name;

}

string get\_surename()

{

return this->surename;

}

string get\_gender()

{

return this->gender;

}

int get\_age()

{

return this->age;

}

};

class subject

{

// Attributes

public:

static string data\_file;

string subject\_name, subject\_id;

int crdit\_hrs;

// GETTERS

void set\_item\_name(string n)

{

this->subject\_name = n;

}

void set\_subject\_id(string i)

{

this->subject\_id = i;

}

void set\_crdit\_hrs(int p)

{

this->crdit\_hrs = p;

}

// SETTERS

string get\_item\_name()

{

return this->subject\_name;

}

string get\_subject\_id()

{

return this->subject\_id;

}

int get\_crdit\_hrs()

{

return this->crdit\_hrs;

}

string get\_data\_file()

{

return this->data\_file;

}

};

class staff\_member : public HOD

{

// Attributes

static string data\_file;

string c\_id;

public:

// GETTERS

subject p;

void set\_staff\_memberid(string s)

{

this->c\_id = s;

}

//SETTERS

string get\_staff\_memberid()

{

return this->c\_id;

}

string get\_data\_file()

{

return this->data\_file;

}

};

string staff\_member ::data\_file = "staff\_member";

string subject ::data\_file = "subject";

class data

{

public:

staff\_member C[100];

subject P[100];

int psize, csize;

data()

{

string c\_id, name, sure, gend, iname, iid;

int pric, age;

ifstream cfile;

cfile.open("staff\_member.txt");

if (cfile)

{

int i = 0;

while (!cfile.eof())

{ // READING staff\_member RECORD FROM DATA FILE staff\_member.TXT

csize = i;

cfile >> c\_id >> name >> sure >> gend >> age >> iid >> iname >> pric;

C[i].set\_age(age);

C[i].set\_staff\_memberid(c\_id);

C[i].set\_gender(gend);

C[i].set\_name(name);

C[i].set\_surname(sure);

C[i].p.set\_subject\_id(iid);

C[i].p.set\_item\_name(iname);

C[i].p.set\_crdit\_hrs(pric);

i++;

}

}

else

cout << "ERROR IN CUSTOMR FILE";

cfile.close();

ifstream pfile;

pfile.open("subjects.txt");

if (pfile)

{

int x = 0;

while (!pfile.eof())

{ // READING subject RECORD FROM DATA FILE subject.TXT

psize = x;

pfile >> iid >> iname >> pric;

P[x].set\_subject\_id(iid);

P[x].set\_item\_name(iname);

P[x].set\_crdit\_hrs(pric);

x++;

}

}

pfile.close();

}

//SHOW subject RECORD

void show\_item()

{

for (int i = 0; i <= psize; i++)

{

cout << P[i].get\_subject\_id() << "\t" << P[i].get\_item\_name() << "\t" << P[i].get\_crdit\_hrs() << endl;

}

}

//SHOW staff\_member RECORD

void show\_staff\_member()

{

for (int i = 0; i <= csize; i++)

{

cout << C[i].get\_age() << "\t" << C[i].gender << "\t" << C[i].get\_staff\_memberid() << "\t" << C[i].get\_gender() << "\t" << C[i].get\_name() << "\t" << C[i].get\_surename() << endl;

}

}

void search\_item()

{

string s;

cout << " ENTER ID ";

cin >> s;

for (int i = 0; i <= psize; i++)

{

if (P[i].get\_subject\_id() == s)

{

cout << P[i].get\_subject\_id() << "\t" << P[i].get\_item\_name() << "\t" << P[i].get\_crdit\_hrs() << endl;

return;

}

}

cout << "Record not found ";

}

void searc\_staff\_member()

{

string s;

cout << " ENTER ID ";

cin >> s;

for (int i = 0; i <= csize; i++)

{

if (C[i].get\_staff\_memberid() == s)

{

cout << P[i].get\_subject\_id() << "\t" << P[i].get\_item\_name() << "\t" << P[i].get\_crdit\_hrs() << endl;

return;

}

}

cout << "Record not found ";

}

void add\_staff\_member()

{

string c\_id, name, sure, gend, iname, iid;

int pric, age;

cout << "Entre staff\_member id ";

cin >> c\_id;

cout << "entre name ";

cin >> name;

cout << "entre surename ";

cin >> sure;

cout << "Entre gender ";

cin >> gend;

cout << "Entre subject name ";

cin >> iname;

cout << "Entre subject id ";

cin >> iid;

cout << "Entre age";

cin >> age;

cout << "Entre subject crdit\_hrs ";

cin >> pric;

csize++;

C[csize].set\_age(age);

C[csize].set\_staff\_memberid(c\_id);

C[csize].set\_gender(gend);

C[csize].set\_name(name);

C[csize].set\_surname(sure);

C[csize].p.set\_subject\_id(iid);

C[csize].p.set\_item\_name(iname);

C[csize].p.set\_crdit\_hrs(pric);

}

void delete\_staff\_member()

{

string s;

cout << "Entre ID";

cin >> s;

int i = 0;

for (i = 0; i <= csize; i++)

{

if (C[i].get\_staff\_memberid() == s)

{

break;

}

}

while (i < csize)

{

C[i] = C[i + 1];

i++;

}

csize--;

}

void delete\_item()

{

string s;

cout << "Entre ID";

cin >> s;

int i = 0;

for (i = 0; i <= psize; i++)

{

if (P[i].get\_subject\_id() == s)

{

break;

}

}

while (i < psize)

{

P[i] = P[i + 1];

i++;

}

psize--;

}

void update\_staff\_member()

{

string c\_id, name, sure, gend, iname, iid;

int pric, age;

string s;

cout << "Entre ID";

cin >> s;

for (int i = 0; i <= csize; i++)

{

if (C[i].get\_staff\_memberid() == s)

{

cout << "Entre staff\_member id ";

cin >> c\_id;

cout << "entre name ";

cin >> name;

cout << "entre surename ";

cin >> sure;

cout << "Entre gender ";

cin >> gend;

cout << "Entre subject name ";

cin >> iname;

cout << "Entre subject id ";

cin >> iid;

cout << "Entre age";

cin >> age;

cout << "Entre subject crdit\_hrs ";

cin >> pric;

C[i].set\_age(age);

C[i].set\_staff\_memberid(c\_id);

C[i].set\_gender(gend);

C[i].set\_name(name);

C[i].set\_surname(sure);

C[i].p.set\_subject\_id(iid);

C[i].p.set\_item\_name(iname);

C[i].p.set\_crdit\_hrs(pric);

return;

}

}

}

void menu()

{

char c;

while (true)

{

system("cls");

cout << "1.ADD staff\_member\n";

cout << "2.Show staff\_member\n";

cout << "3.Show subject\n";

cout << "4.Search staff\_member\n";

cout << "5.Seach subject\n";

cout << "6.Delete staff\_member\n";

cout << "7.Delete subject\n";

cout << "8.Update staff\_member\n";

cout << "9.EXIT\n";

cout << "Entre choice ";

cin >> c;

switch (c)

{

case '1':

system("cls");

this->add\_staff\_member();

system("pause");

break;

case '2':

system("cls");

this->show\_staff\_member();

system("pause");

break;

case '3':

system("cls");

this->show\_item();

system("pause");

break;

case '4':

system("cls");

this->searc\_staff\_member();

system("pause");

break;

case '5':

system("cls");

this->search\_item();

system("pause");

break;

case '6':

system("cls");

this->delete\_staff\_member();

system("pause");

break;

case '7':

system("cls");

this->delete\_item();

system("pause");

break;

case '8':

system("cls");

this->update\_staff\_member();

system("pause");

break;

case '9':

exit(1);

default:

break;

}

}

}

};

int main()

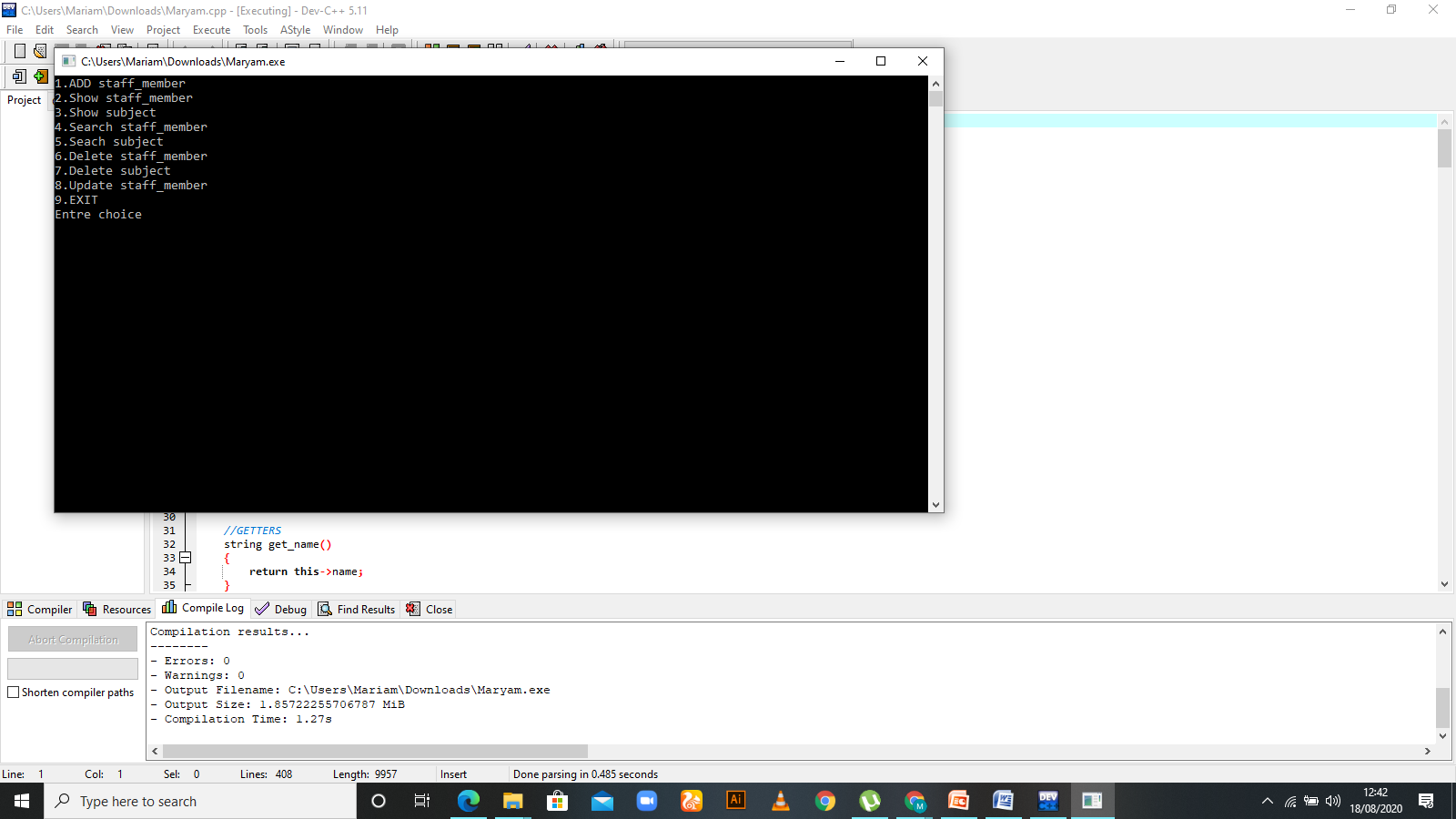
{

data D;

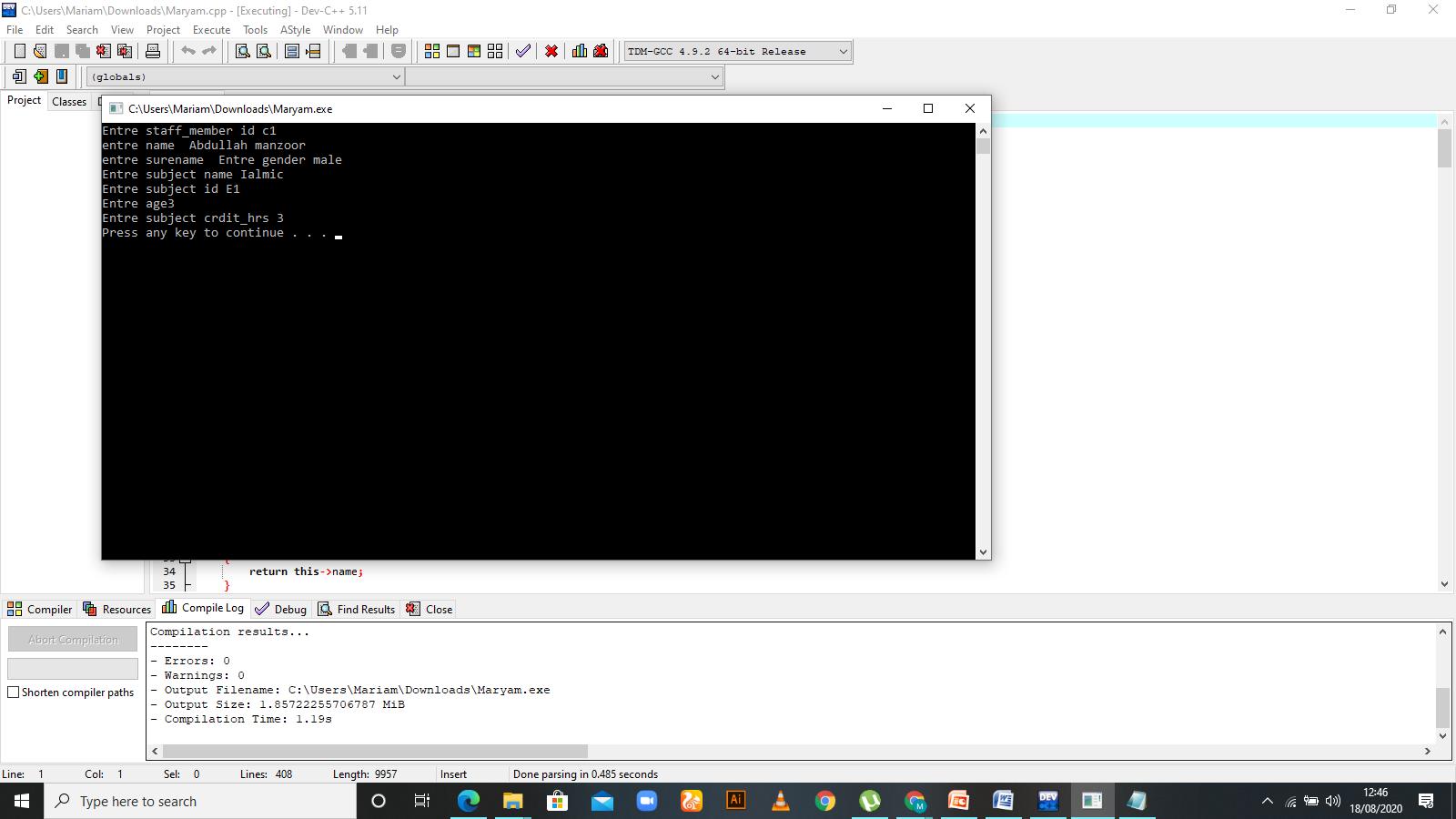
D.menu();

}

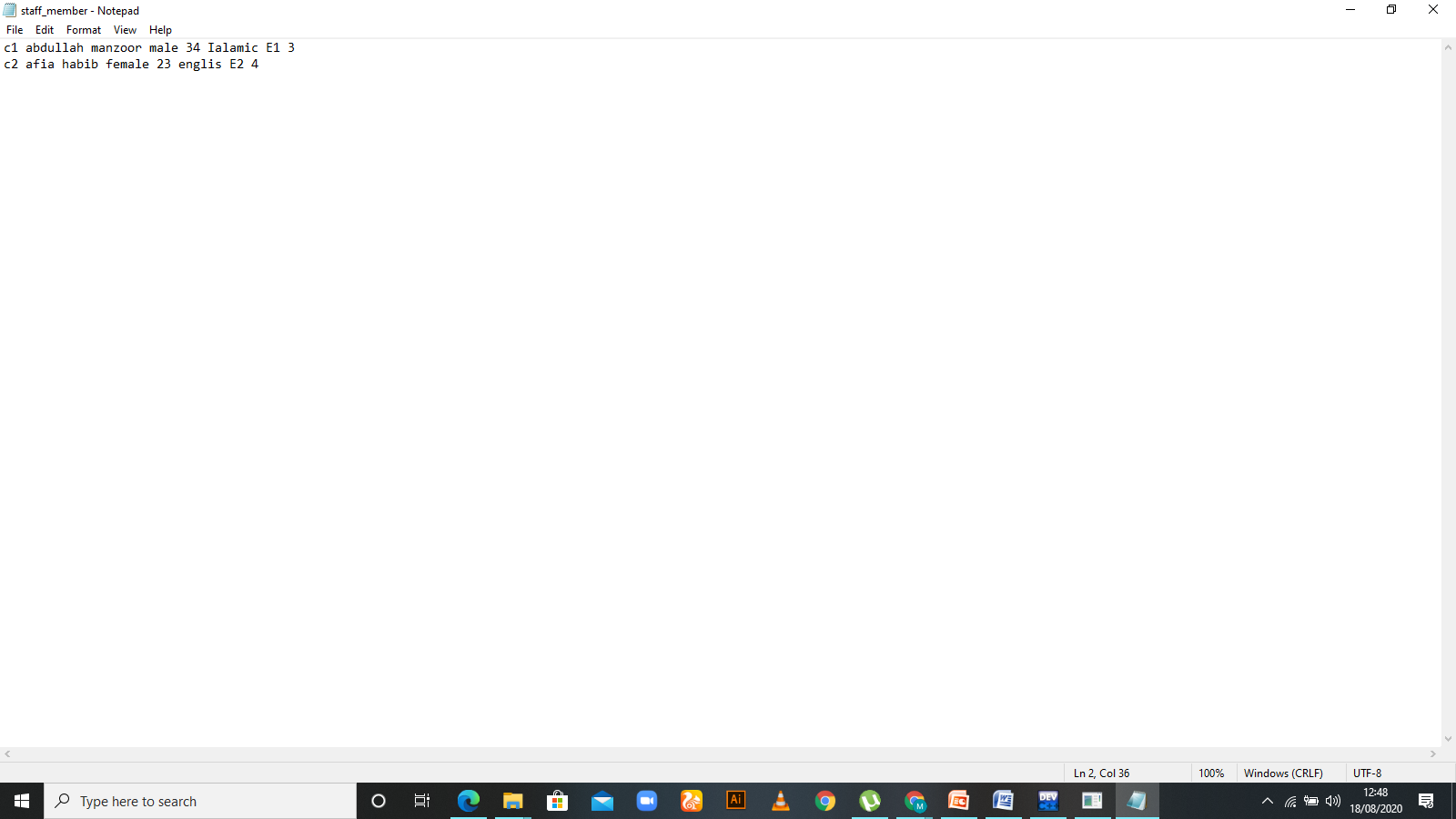
**Output**



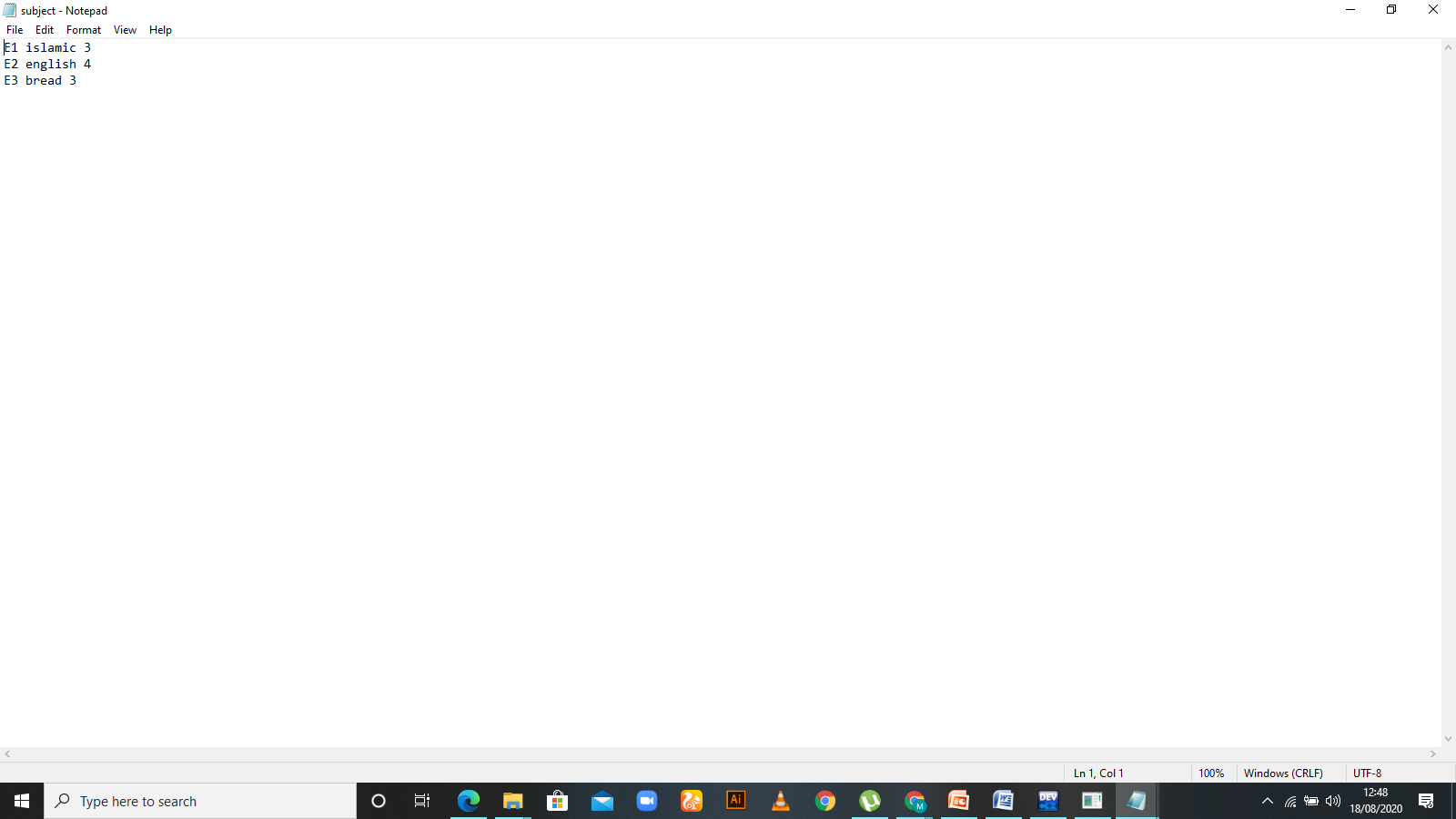
**Output-2**



Staff-Member



**Subject File**



Github Link

<https://github.com/Mariam-mk/paper-18321519-085/upload>