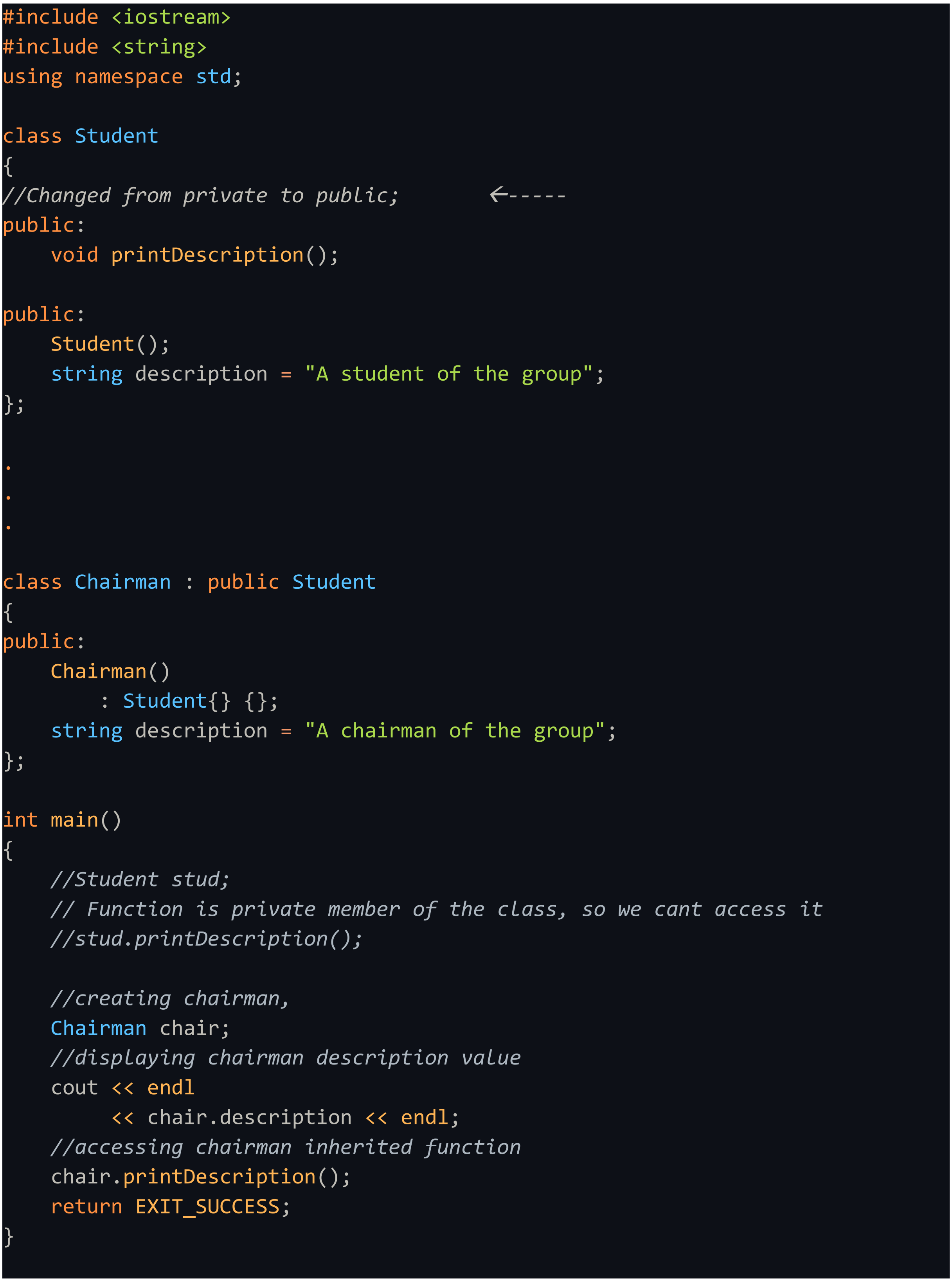
EIT, 306387, Gabriel Ćwiek, MAPT 2, lab 1, https://github.com/Astaree/MAPT_2

1. Ex. 1: Find why code doesn’t work, make chairman child of student, create new chairman and display vales.



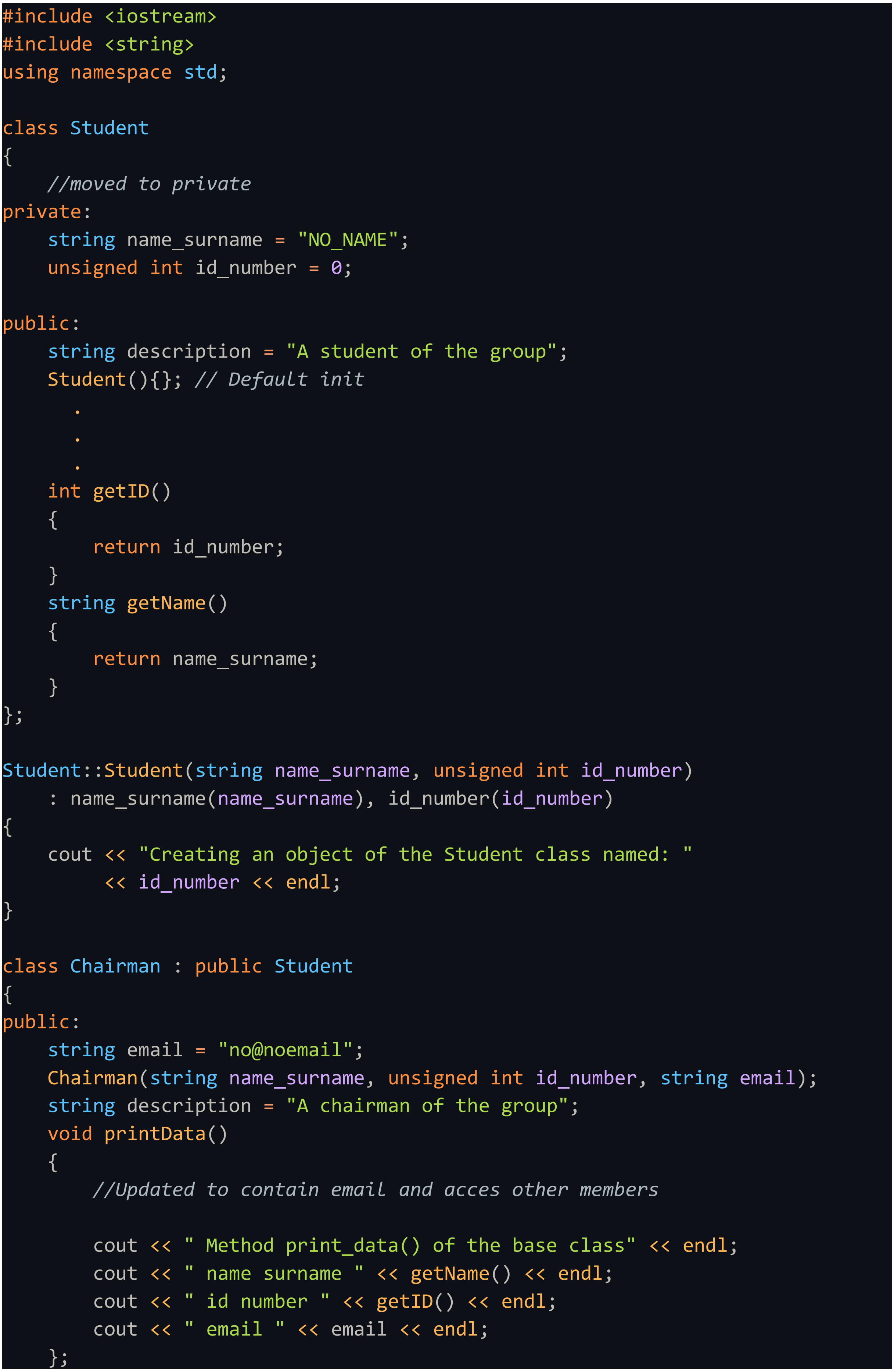
```
#include <iostream>
#include <string>
using namespace std;

class Student
{
//Changed from private to public;        ←-----
public:
    void printDescription();

public:
    Student();
    string description = "A student of the group";
};

.
.
.

class Chairman : public Student
{
public:
    Chairman()
        : Student{} {};
    string description = "A chairman of the group";
};

int main()
{
    //Student stud;
    // Function is private member of the class, so we cant access it
    //stud.printDescription();

    //creating chairman,
    Chairman chair;
    //displaying chairman description value
    cout << endl
         << chair.description << endl;
    //accessing chairman inherited function
    chair.printDescription();
    return EXIT_SUCCESS;
}
```

```
Output:

Creating an object of the class Student, with description:
 A student of the group

A chairman of the group
description: A student of the group
```

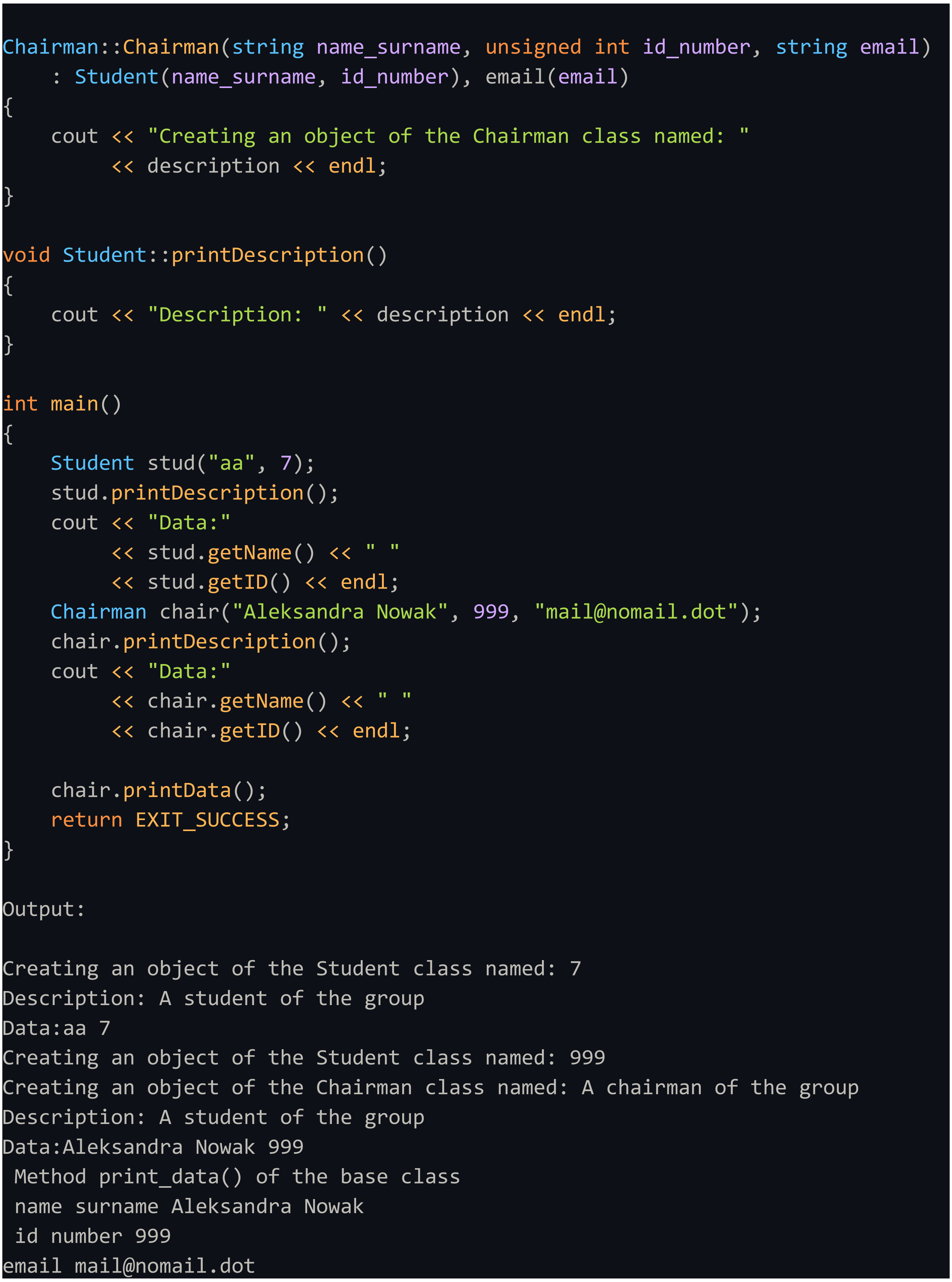
2. Ex. 2: Update chairman function printdescription():

```
.
.
.
class Chairman : public Student
{
public:
    Chairman()
        : Student{} {};
    string description = "A chairman of the group";
    public:
    void printDescription();
};

void Chairman::printDescription()
{
    cout << "description: " << description << endl;
}

int main()
{
    …
    Chairman chair;
    //displaying chairman description value
    cout << endl
         << chair.description << endl;
    //accessing chairman inherited function
    chair.printDescription();
    return EXIT_SUCCESS;
}

Output:
Creating an object of the class Student, with description:
 A student of the group

A chairman of the group
description: A chairman of the group
```

3. Ex. 3:



```
#include <iostream>
#include <string>
using namespace std;

class Student
{
    //moved to private
private:
    string name_surname = "NO_NAME";
    unsigned int id_number = 0;

public:
    string description = "A student of the group";
    Student(){}; // Default init
      .
      .
      .
    int getID()
    {
        return id_number;
    }
    string getName()
    {
        return name_surname;
    }
};

Student::Student(string name_surname, unsigned int id_number)
    : name_surname(name_surname), id_number(id_number)
{
    cout << "Creating an object of the Student class named: "
         << id_number << endl;
}

class Chairman : public Student
{
public:
    string email = "no@noemail";
    Chairman(string name_surname, unsigned int id_number, string email);
    string description = "A chairman of the group";
    void printData()
    {
        //Updated to contain email and acces other members

        cout << " Method print_data() of the base class" << endl;
        cout << " name surname " << getName() << endl;
        cout << " id number " << getID() << endl;
        cout << " email " << email << endl;
    };
```



```
Chairman::Chairman(string name_surname, unsigned int id_number, string email)
    : Student(name_surname, id_number), email(email)
{
    cout << "Creating an object of the Chairman class named: "
         << description << endl;
}

void Student::printDescription()
{
    cout << "Description: " << description << endl;
}

int main()
{
    Student stud("aa", 7);
    stud.printDescription();
    cout << "Data:"
         << stud.getName() << " "
         << stud.getID() << endl;
    Chairman chair("Aleksandra Nowak", 999, "mail@nomail.dot");
    chair.printDescription();
    cout << "Data:"
         << chair.getName() << " "
         << chair.getID() << endl;

    chair.printData();
    return EXIT_SUCCESS;
}

Output:

Creating an object of the Student class named: 7
Description: A student of the group
Data:aa 7
Creating an object of the Student class named: 999
Creating an object of the Chairman class named: A chairman of the group
Description: A student of the group
Data:Aleksandra Nowak 999
 Method print_data() of the base class
 name surname Aleksandra Nowak
 id number 999
email mail@nomail.dot
```