# COMPUTER PROGRAMMİNG 2 TERM PROJECT REPORT

## 1-SUMMARY

This document was created to explain the solution for the Computer Programmin 2 Term Project.The project was written in Java Programming Language in Apache Netbeans 12.3 and Swing framework is used as an interface.Derby 10.14 version is used as database.The document contains the subject of the project, its description, work flow chart, project outputs, what has been done during the project process, annotations and bibliography.

## 2-PROJECT TOPİC

The subject of the project is personnel tracking automation. Employees' id, name, surname, e-mail, password, department, salary, entry and exit times, on which days they are on leave and how many times they are late for work

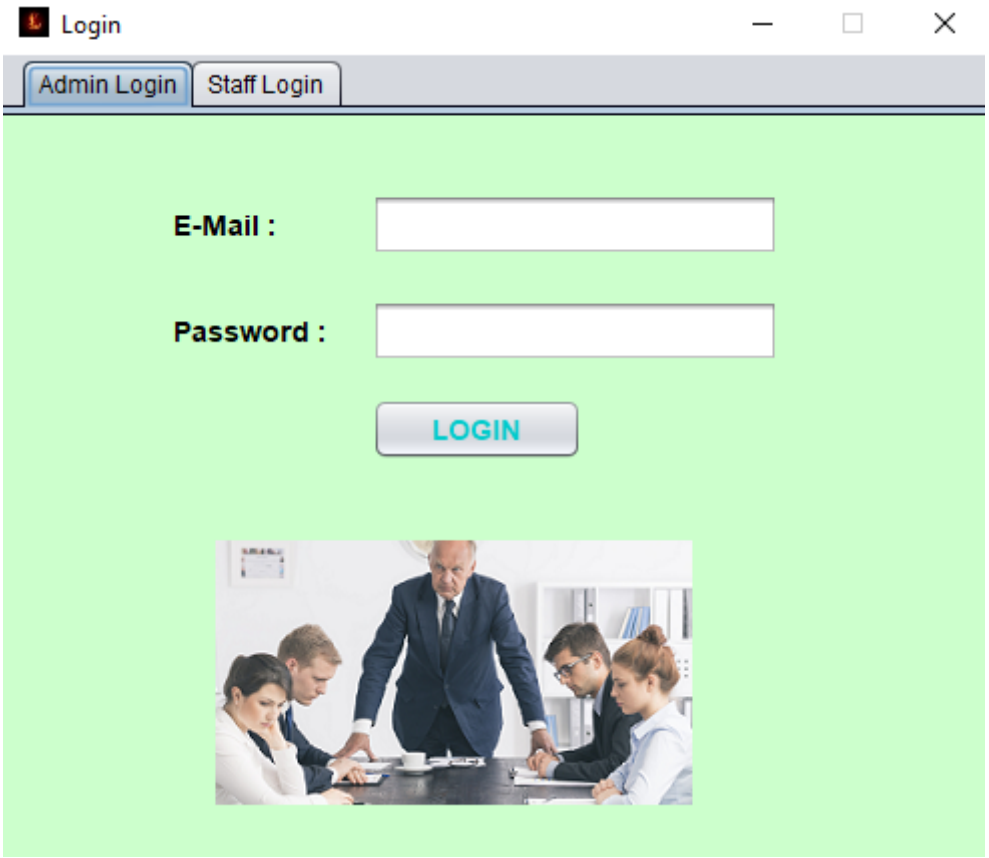
## 3-PROJECT WORK FLOW CHART

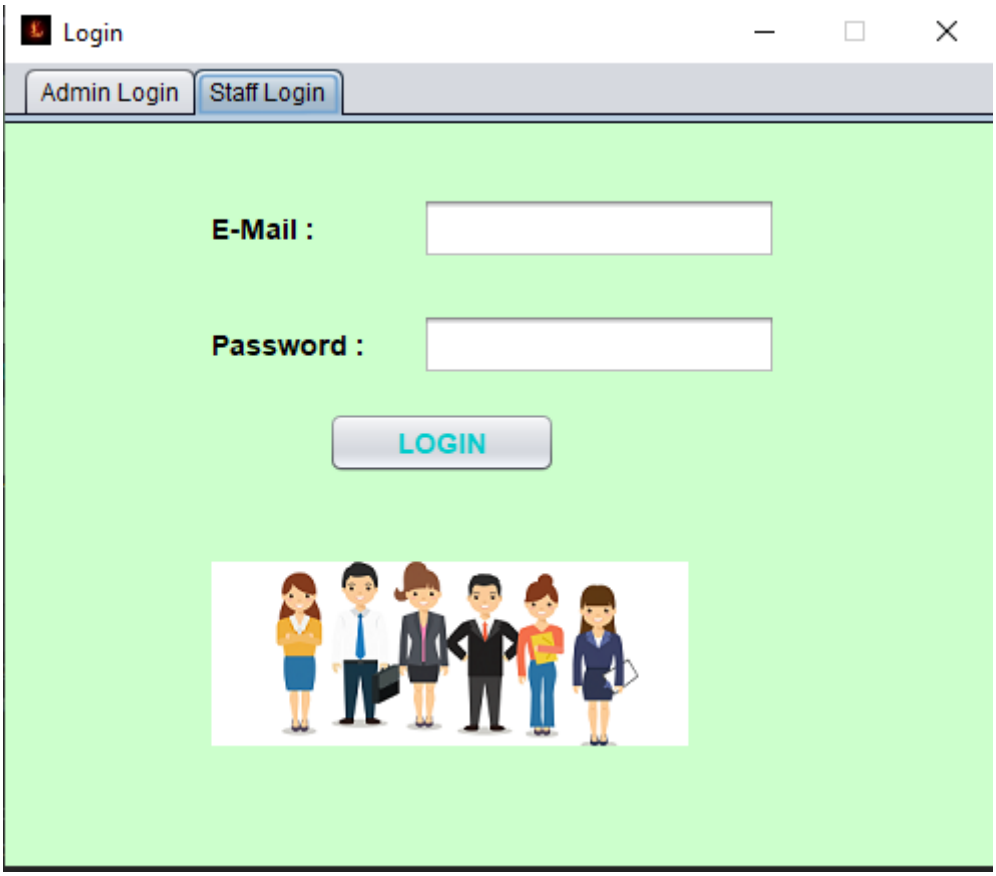
| The Work Done | 28.04-05.05 | 05.05-12.05 | 12.05-19.05 | 19.05-26.05 |
|---|---|---|---|---|
| **Research and Decision** | | | | |
| **Design and Code Writing** | | | | |
| **Test Phase** | | | | |
| **Report Generation** | | | | |

## 4-PROJECT DESIGN

There are 3 different database tables and 4 different JFrames in the project.Personnel database table contains general information about personnel.Yoneticiler database table contains information about admins and Ekstra database table contains the number of late arrivals and days off of the staff.Personel_Screen JFrame contains staff sending requests for leave to managers. After logging in, the staff is directed to this page and informs the manager between which dates they want to get permission.Login_Screen Jframe contains admin and staff login. Depending on the type of person logged in, the administrator and staff accessing the required pages are directed to the required pages. Main_Screen JFrame contains general information abouts personnel to the database. The administrator enters the characteristics of the personnel to be added and records them in the database.Admin_Screen JFrame contains the administrator to allow staff and record the days when they are late in the database.Database class contains literal general properties about the database.Personnel class contains features of staff.Personnel_Operations class contains database operations and sql queries.

## 5-PROJECT OUTPUTS AND SUCCES CRITERIA






Main

Menu

Information Bar

ID : 9

Name : Emirhan

Surname : Yar

E-Mail : emrhn@gmail.com

Password : ******

Departmant : İnsan Kaynakları

Salary : 2500

CHECH-IN TIME : 10:00

CHECH-OUT TIME : 18:00

CLEAN

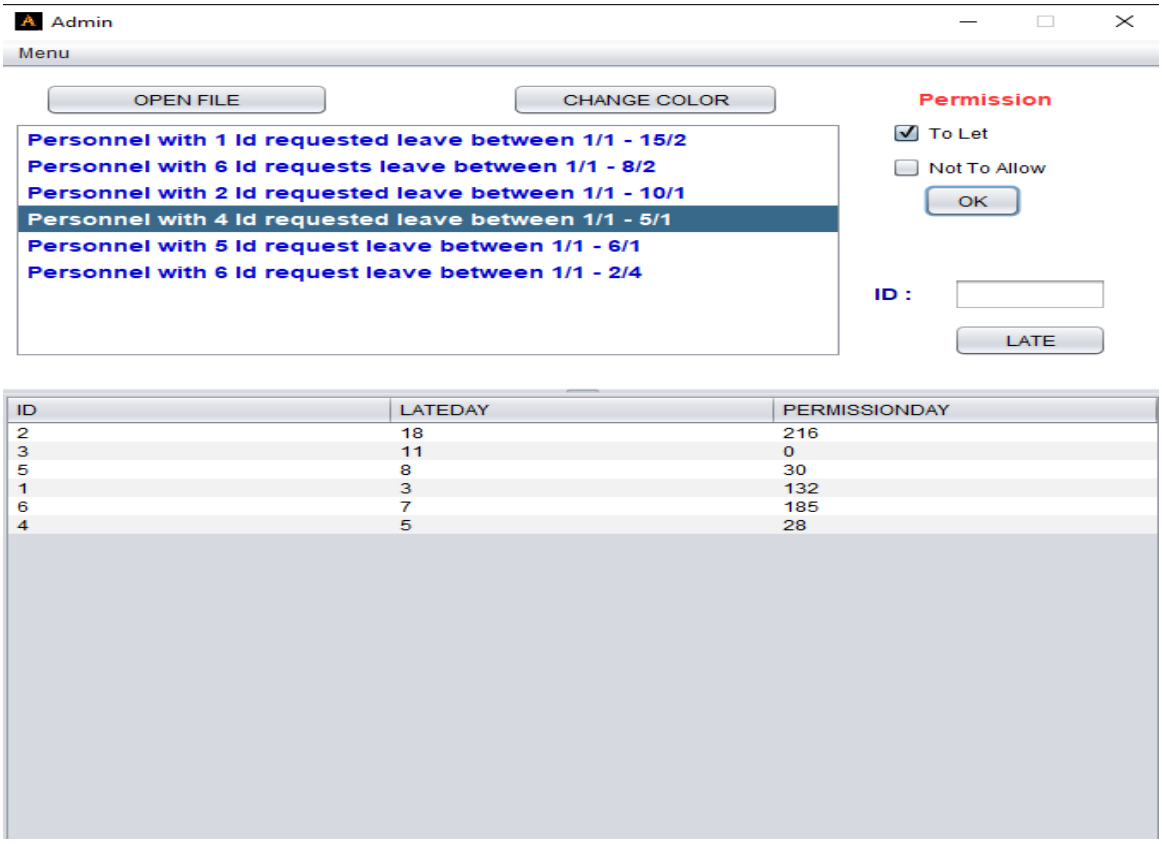
ADD STAFF

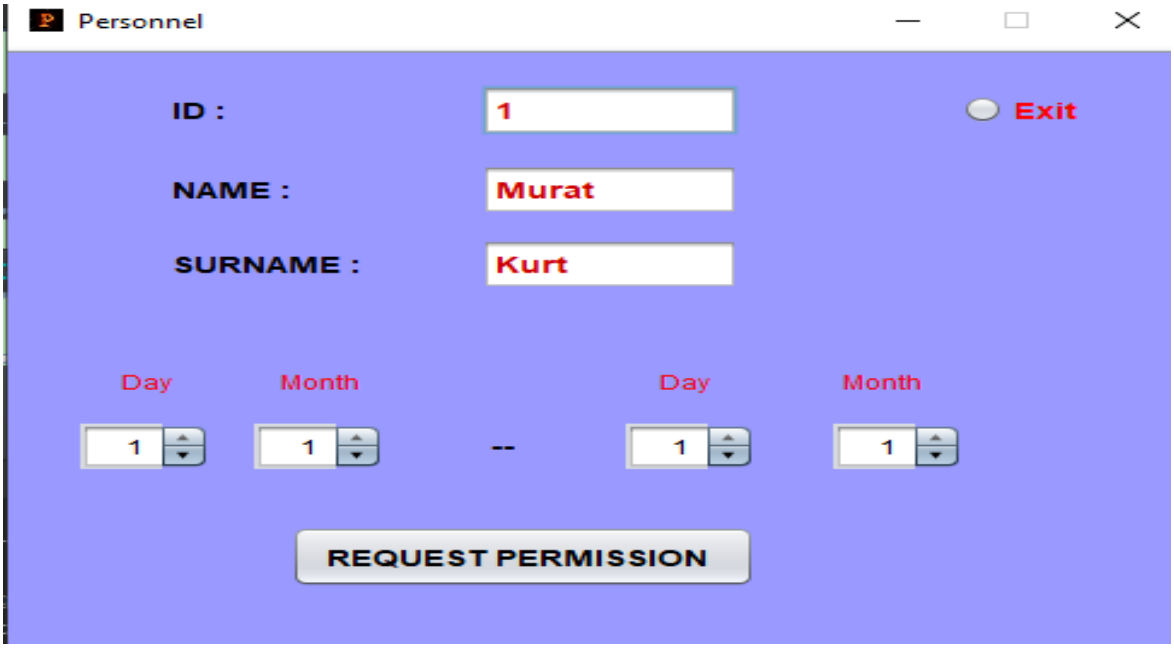
DELETE STAFF

UPDATE STAFF

Search Bar

| ID | NAME | SURNAME | EMAIL | PASSWORD | DEPARTMANT | SALARY | CHECKIN | CHECKOUT |
|---|---|---|---|---|---|---|---|---|
| 1 | Murat | Kurt | murat@gmai... | 123456 | Bilişim | 3000 | 09:00 | 17:00 |
| 4 | Eren | Takır | eren@hotma... | rty789 | Bilişim | 2500 | 09:00 | 17:00 |
| 5 | Osman | Alan | osman@gm... | vbn456 | İnsan Kaynak... | 3500 | 10:00 | 18:00 |
| 6 | Berk | Aslan | berk@outloo... | okl789 | Bilişim | 4000 | 09:00 | 17:00 |
| 7 | Mustafa | Dege | mustafa@g... | lkj123 | Pazarlama | 4000 | 11:00 | 17:00 |
| 8 | Serdar | Kaplan | serdar@hot... | ghj456 | Bilişim | 3000 | 09:00 | 17:00 |
| 9 | Emirhan | Yar | emrhn@gma... | fbg159 | İnsan Kaynak... | 2500 | 10:00 | 18:00 |
| 2 | Can | As | can@gmail.c... | asd478 | Bilişim | 3500 | 09:00 | 17:00 |
| 3 | Ilhan | Balcı | ilhan@outloo... | fgh369 | Insan Kaynak... | 3000 | 10:00 | 18:00 |
| 10 | Semih | Şen | semih@gma... | ghf451 | Bilişim | 4000 | 09:00 | 17:00 |
| 11 | Musa | Yıldız | musa@gmai... | gbk419 | Pazarlama | 6000 | 11:00 | 17:00 |
| 12 | Samet | Taş | samet@outlo... | ygn594 | İnsanKaynakl... | 4000 | 10:00 | 18:00 |
| 13 | Bilal | Kartal | bilal@gmail.... | asg162 | Bilişim | 5000 | 09:00 | 17:00 |
| 14 | Çağatay | Akın | cagatay@gm... | fbg658 | Pazarlama | 4000 | 11:00 | 17:00 |


Admin
Menu
OPEN FILE
CHANGE COLOR
Permission
To Let
Not To Allow
OK
Personnel with 1 Id requested leave between 1/1 - 15/2
Personnel with 6 Id requests leave between 1/1 - 8/2
Personnel with 2 Id requested leave between 1/1 - 10/1
Personnel with 4 Id requested leave between 1/1 - 5/1
Personnel with 5 Id request leave between 1/1 - 6/1
Personnel with 6 Id request leave between 1/1 - 2/4
ID :
LATE
ID LATEDAY PERMISSIONDAY
2 18 216
3 11 0
5 8 30
1 3 132
6 7 185
4 5 28


Personnel
ID :
1
Exit
NAME :
Murat
SURNAME :
Kurt
Day
Month
Day
Month
1
1
--
1
1
REQUEST PERMISSION

## 6-ACTIONS DURING THE PROJECT

- First of all, I thought and noted what I could do according to the subject of the project.
- Then creating the personnel class and defining the characteristics of the personnel and I created a database in accordance with these features.
- I added managers and personnel to the database. I created interfaces for administrative and personnel entries and checked these entries with the information in the database.
- Then I created the interface where the information of the personnel in the database is transferred, I added the necessary components and codes.
- Then I designed an interface and added the necessary components for the staff to ask for permission from the management after staff entry.
- Finally, I created the interface where the administrator saw the permission requests accordingly or not and saved the late staff to the database and added the necessary compenents.

## 7-ADDITIONAL EXPLANATIONS

The places that took the most time while doing the project were placing the components in the interface actually this job is very simple but takes time. I made researches on various sites on the internet to add data, delete, update, with different queries to the database. I also did the research of adding necessary events to the compenents to check the wrong entries of the user and I designed the program not to be broken against the wrong inputs of the user. I also found the regular expressions required for these incorrect entries and added them to the program.I added logos with Jframe as a last and extra. Since I could not find a place where I could use generics in the project, I researched and learned this subject separately and added it to the report by making an example. Apart from that, I did all the issues that should be in the project

## 8-REFERENCES


https://stackoverflow.com/questions/27815400/retrieving-data-from-jdbc-database-into-jtable

https://www.udemy.com/course/sifirdan-ileri-seviyeye-komple-java-gelistirici-kursu/learn/lecture/8596948?start=195#content

https://www.javatpoint.com/java-string-format

https://regexr.com

https://www.w3schools.com/sql/

# GENERİCS USAGE

It is the type of class that keeps all reference data types in it, and we can decide which reference type to keep and do the same operations on it.

The first advantage of generics in Java is to ensure that any type incompatibilities that may occur can be detected and corrected during compile. The second advantage is that there is no need for casting. The third advantage is that it prevents code repetition by writing code that can work with different types at the same time.

### Generic Class Example

Yazdirma Sınıfı is a generic class.Encapsulates the incoming parameter of the Character, String, Integer and Student class type <E> and acts as a student and the program runs smoothly.

```
public class YazdirmaSinifi<E> {

    public void yazdir(E[] dizi) {

        for (E e : dizi) {

            System.out.println(e);

        }

    }

}
```

```
Character[] char_dizi = {'J', 'A', 'V', 'A'};
Integer[] integer_dizi = {1, 2, 3, 4, 5, 6};
String[] string_dizi = {"Java", "Python", "C++", "Php"};
Ogrenci[] ogrenci_dizi = {new Ogrenci("Özkan"), new Ogrenci("Mehmet"), new Ogrenci("Ahmet")};

YazdirmaSinifi<Character> yazdir_char = new YazdirmaSinifi<Character>();
YazdirmaSinifi<String> yazdir_string = new YazdirmaSinifi<String>();
YazdirmaSinifi<Integer> yazdir_int = new YazdirmaSinifi<Integer>();
YazdirmaSinifi<Ogrenci> yazdir_ogrenci = new YazdirmaSinifi<Ogrenci>();
```

### Generic Method Example

The data type sent to every place with E in the method is written and the program runs smoothly according to it.

```
    public static <E> void yazdir(E[] dizi) {
        for (E e : dizi) {

            System.out.println(e);

        }

    }

}
```