

# Email Administrator

```
import java.util.*;
import java.io.*;

class Email
{
    //Scanner class global
    public Scanner s = new Scanner(System.in);

    //Setting variables as private to not give access directly
    private String fname;
    private String lname;
    private String dept;
    private String email;
    private String password;
    private int mailCapacity = 500;
    private String alterEmail;

    //Constructor to receive firstname ,lastname
    public Email(String fname,String lname)
    {
        this.fname = fname;
        this.lname=lname;
        System.out.println("New Employee:"+ this.fname+" "+ this.lname);

        //calling methods
        this.dept = this.setDept(); //department
        this.password = this.generatePassword(8); //password
        this.email = this.generateEmail();//email generation
    }

    //Asking for dept method
    private String setDept()
    {
        System.out.println("Department codes \n1. For sales \n2. For
Development \n3. For Accounting \n4. None");
        boolean flag = false;
        do
        {
            System.out.println("Enter Department code");
            int choice = s.nextInt();
            switch(choice)
            {
                case 1 :
                    return "Sales";
                case 2 :
```

# Email Administrator

```
        return "Development";
    case 3 :
        return "Accounting";
    case 4 :
        return "None";
    default:
        System.out.println("Invalid choice please choose it
again");
    }
}while(!flag);
return null;
}

//Generate Random password method
private String generatePassword(int length)
{
    Random r = new Random();
    String CapitalChars = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
    String SmallChars = "abcdefghijklmnopqrstuvwxyz";
    String numbers = "0123456789";
    String symbols = "!@#$%&?";
    String values = CapitalChars+SmallChars+numbers+symbols;
    String password="";
    for(int i=0;i<length;i++)
    {
        password = password+values.charAt(r.nextInt(values.length()));
    }
    return password;
}

//Generate mail method
private String generateEmail()
{
    return
this.fname.toLowerCase()+"."+this.lname.toLowerCase()+"@"+this.dept.toLowerCase()
e()+".com";
}

//Change password method
public void setPassword()
{
    boolean flag = false;
    do
    {
```

# Email Administrator

```
System.out.println("Do you want to change your password!(Y/N)");
char choice = s.next().charAt(0);
if(choice == 'Y' || choice == 'y')
{
    flag = true;
    System.out.println("Enter the current password");
    String temp = s.next();
    if(temp.equals(this.password))
    {
        System.out.println("Enter new password:");
        this.password = s.next();
        System.out.println("Password changed successfully");
    }
    else
    {
        System.out.println("Incorrect password");
        System.out.println("Enter the current password");
        String temp1 = s.next();

        if(temp1.equals(this.password))
        {
            System.out.println("Enter new password:");
            this.password = s.next();
            System.out.println("Password changed successfully");
        }
        else
        {
            System.out.println("Incorrect password");
        }
    }
}
else if(choice == 'N' || choice == 'n')
{
    flag = true;
    System.out.println("password changed option cancelled!");
}
else
{
    System.out.println("Enter valid choice");
}
}while(!flag);
}
```

# Email Administrator

```
//set mailbox capacity method
public void setMailCap()
{
    System.out.println("Current capacity =" + this.mailCapacity + "mb");
    System.out.println("Enter new mailbox capacity:");
    this.mailCapacity = s.nextInt();
    System.out.println("Mailbox Capacity is successfully changed");
}

// set alternate email method
public void alternateEmail()
{
    System.out.println("Enter new alternate mail:");
    this.alterEmail = s.next();
    System.out.println("Alternate email is set");
}

//Display user information method
public void getInfo()
{
    System.out.println("New Employee:" + this.fname + " " + this.lname);
    System.out.println("Department:" + this.dept);
    System.out.println("Email:" + this.email);
    System.out.println("Password:" + this.password);
    System.out.println("Mailbox Capacity:" + this.mailCapacity + "mb");
    System.out.println("Alternate e-mail:" + this.alterEmail);
}
}

public class EmailApp
{
    public static void main(String[] args)
    {
        Scanner s1 = new Scanner(System.in);

        //User info
        System.out.print("Enter first name : ");
        String f_name = s1.next();
        System.out.print("Enter last name : ");
        String l_name = s1.next();

        //creating object for email class
        Email em1 = new Email(f_name, l_name);
    }
}
```

# Email Administrator

```
int choice = -1;
do
{
    System.out.println("\n*****\nEnter your choice \n1.
Show Information \n2. Change password \n3. Change mailbox capacity \n4. Set
alternate email \n5. Exit");
    choice = s1.nextInt();
    switch(choice)
    {
        case 1 :
            em1.getInfo();
            break;
        case 2 :
            em1.setPassword();
            break;
        case 3 :
            em1.setMailCap();
            break;
        case 4 :
            em1.alternateEmail();
            break;
        case 5 :
            System.out.println("Thank You for using Email
Application");
            break;
        default:
            System.out.println("Invalid choice!!\n Enter proper choice
again..");
            break;
    }
}while(choice!=5);
}
```

# Email Administrator

From notepad

```
import java.util.*;
```

```
import java.io.*;
```

```
class Email
```

```
{
```

```
    //Scanner class global
```

```
    public Scanner s = new Scanner(System.in);
```

```
    //Setting variables as private to not give access directly
```

```
    private String fname;
```

```
    private String lname;
```

```
    private String dept;
```

```
    private String email;
```

```
    private String password;
```

```
    private int mailCapacity = 500;
```

```
    private String alterEmail;
```

```
    //Constructor to receive firstname ,lastname
```

```
    public Email(String fname,String lname)
```

```
    {
```

```
        this.fname = fname;
```

```
        this.lname=lname;
```

```
        System.out.println("New Employee:" + this.fname+" "+ this.lname);
```

```
    //calling methods
```

```
        this.dept = this.setDept(); //department
```

```
        this.password = this.generatePassword(8); //password
```

```
        this.email = this.generateEmail();//email generation
```

```
    }
```

# Email Administrator

```
//Asking for dept method

private String setDept()
{
    System.out.println("Department codes \n1. For sales \n2. For Development \n3. For Accounting \n4.
None");

    boolean flag = false;

    do
    {
        System.out.println("Enter Department code");

        int choice = s.nextInt();

        switch(choice)
        {
            case 1 :
                return "Sales";

            case 2 :
                return "Development";

            case 3 :
                return "Accounting";

            case 4 :
                return "None";

            default:
                System.out.println("Invalid choice please choose it again");

        }
    }while(!flag);

    return null;
}
```

```
//Generate Random password method

private String generatePassword(int length)
{
    Random r = new Random();
```

# Email Administrator

```
String CapitalChars = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
String SmallChars = "abcdefghijklmnopqrstuvwxyz";
String numbers = "0123456789";
String symbols = "!@#$%&?";
String values = CapitalChars+SmallChars+numbers+symbols;
String password="";
for(int i=0;i<length;i++)
{
    password = password+values.charAt(r.nextInt(values.length()));
}
return password;
}

//Generate mail method
private String generateEmail()
{
    return this.fname.toLowerCase()+"."+this.lname.toLowerCase()+"@"+this.dept.toLowerCase()+".com";
}

//Change password method
public void setPassword()
{
    boolean flag = false;
    do
    {
        System.out.println("Do you want to change your password!(Y/N)");
        char choice = s.next().charAt(0);
        if(choice == 'Y' || choice == 'y')
        {
            flag = true;
        }
    }
}
```



# Email Administrator

```
System.out.println("Enter the current password");

String temp = s.next();

if(temp.equals(this.password))
{
    System.out.println("Enter new password:");
    this.password = s.next();
    System.out.println("Password changed successfully");
}
else
{
    System.out.println("Incorrect password");
    System.out.println("Enter the current password");
    String temp1 = s.next();

    if(temp1.equals(this.password))
    {
        System.out.println("Enter new password:");
        this.password = s.next();
        System.out.println("Password changed successfully");
    }
    else
    {
        System.out.println("Incorrect password");

    }
}

}

else if(choice == 'N' || choice == 'n')
{
    flag = true;
```

# Email Administrator

```
        System.out.println("password changed option cancelled!");
    }
    else
    {
        System.out.println("Enter valid choice");
    }
}while(!flag);
}

//set mailbox capacity method
public void setMailCap()
{
    System.out.println("Current capacity =" + this.mailCapacity + "mb");
    System.out.println("Enter new mailbox capacity:");
    this.mailCapacity = s.nextInt();
    System.out.println("Mailbox Capacity is successfully changed");
}

// set alternate email method
public void alternateEmail()
{
    System.out.println("Enter new alternate mail:");
    this.alterEmail = s.next();
    System.out.println("Alternate email is set");
}

//Display user information method
public void getInfo()
{
    System.out.println("New Employee:" + this.fname + " " + this.lname);
    System.out.println("Department:" + this.dept);
}
```

# Email Administrator

```
        System.out.println("Email:" + this.email);

        System.out.println("Password:" + this.password);

        System.out.println("Mailbox Capacity:" + this.mailCapacity+"mb");

        System.out.println("Alternate e-mail:" + this.alterEmail);

    }

}

public class EmailApp
{
    public static void main(String[] args)
    {
        Scanner s1 = new Scanner(System.in);

        //User info
        System.out.print("Enter first name : ");

        String f_name = s1.next();

        System.out.print("Enter last name : ") ;

        String l_name = s1.next();

        //creating object for email class
        Email em1 = new Email(f_name,l_name);

        int choice = -1;

        do
        {
            System.out.println("\n*****\nEnter your choice \n1. Show Information \n2. Change
password \n3. Change mailbox capacity \n4. Set alternate email \n5. Exit");

            choice = s1.nextInt();

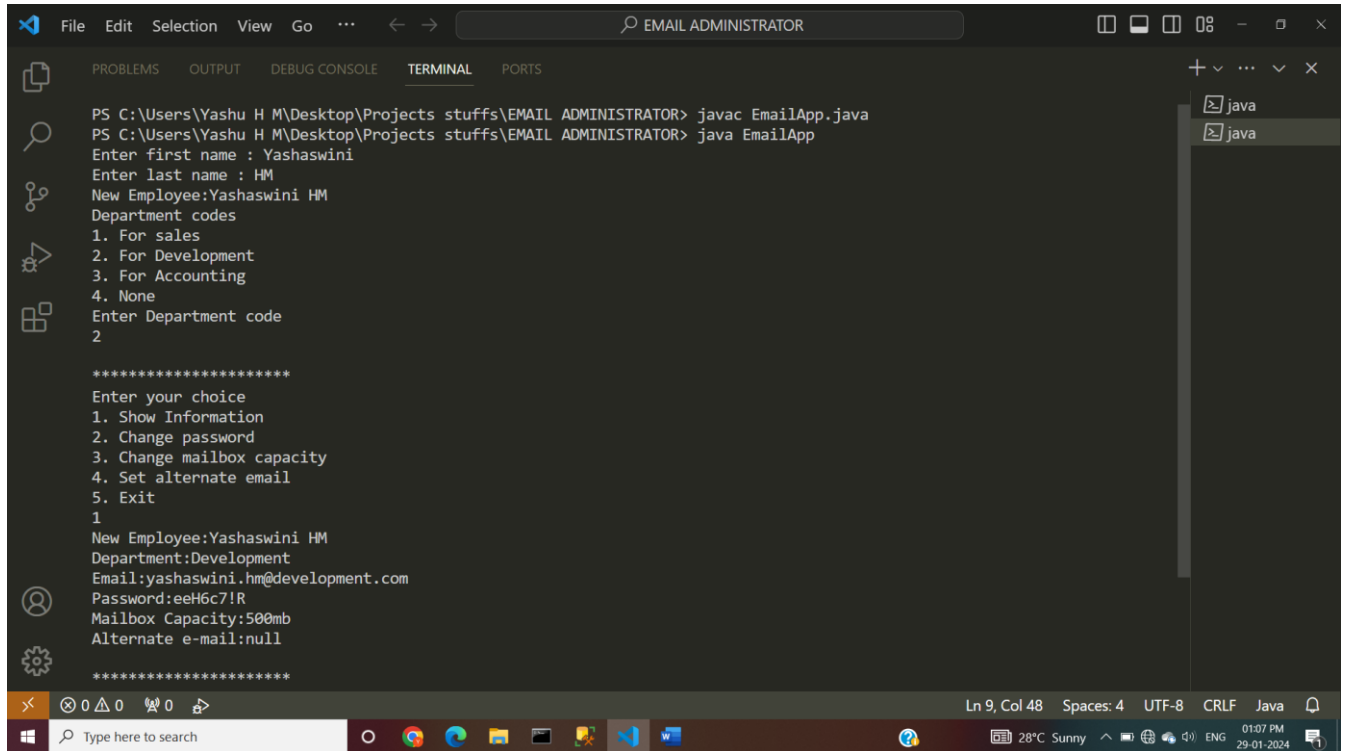
            switch(choice)
            {
```

# Email Administrator

```
case 1 :  
    em1.getInfo();  
    break;  
case 2 :  
    em1.setPassword();  
    break;  
case 3 :  
    em1.setMailCap();  
    break;  
case 4 :  
    em1.alternateEmail();  
    break;  
case 5 :  
    System.out.println("Thank You for using Email Application");  
    break;  
default:  
    System.out.println("Invalid choice!!\n Enter proper choice again..");  
    break;  
}  
}while(choice!=5);  
}  
}
```

# Email Administrator

## OUTPUT 1 : Taking Employee's Credentials and Information viewed

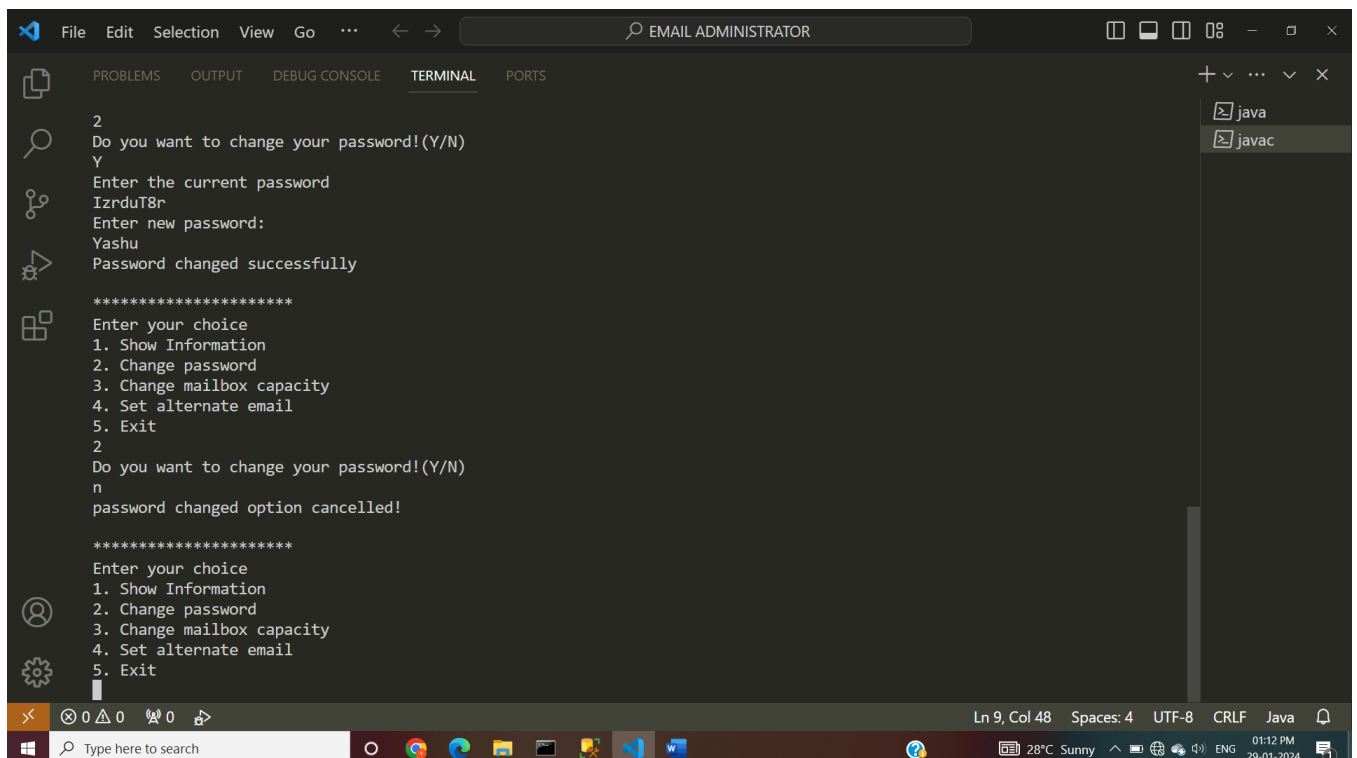


```
PS C:\Users\Yashu H M\Desktop\Projects stuffs\EMAIL ADMINISTRATOR> javac EmailApp.java
PS C:\Users\Yashu H M\Desktop\Projects stuffs\EMAIL ADMINISTRATOR> java EmailApp
Enter first name : Yashaswini
Enter last name : HM
New Employee:Yashaswini HM
Department codes
1. For sales
2. For Development
3. For Accounting
4. None
Enter Department code
2

*****
Enter your choice
1. Show Information
2. Change password
3. Change mailbox capacity
4. Set alternate email
5. Exit
1
New Employee:Yashaswini HM
Department:Development
Email:yashaswini.hm@development.com
Password:eeH6c7!R
Mailbox Capacity:500mb
Alternate e-mail:null

*****
```

## OUTPUT 2: Changing password (Yes and No)



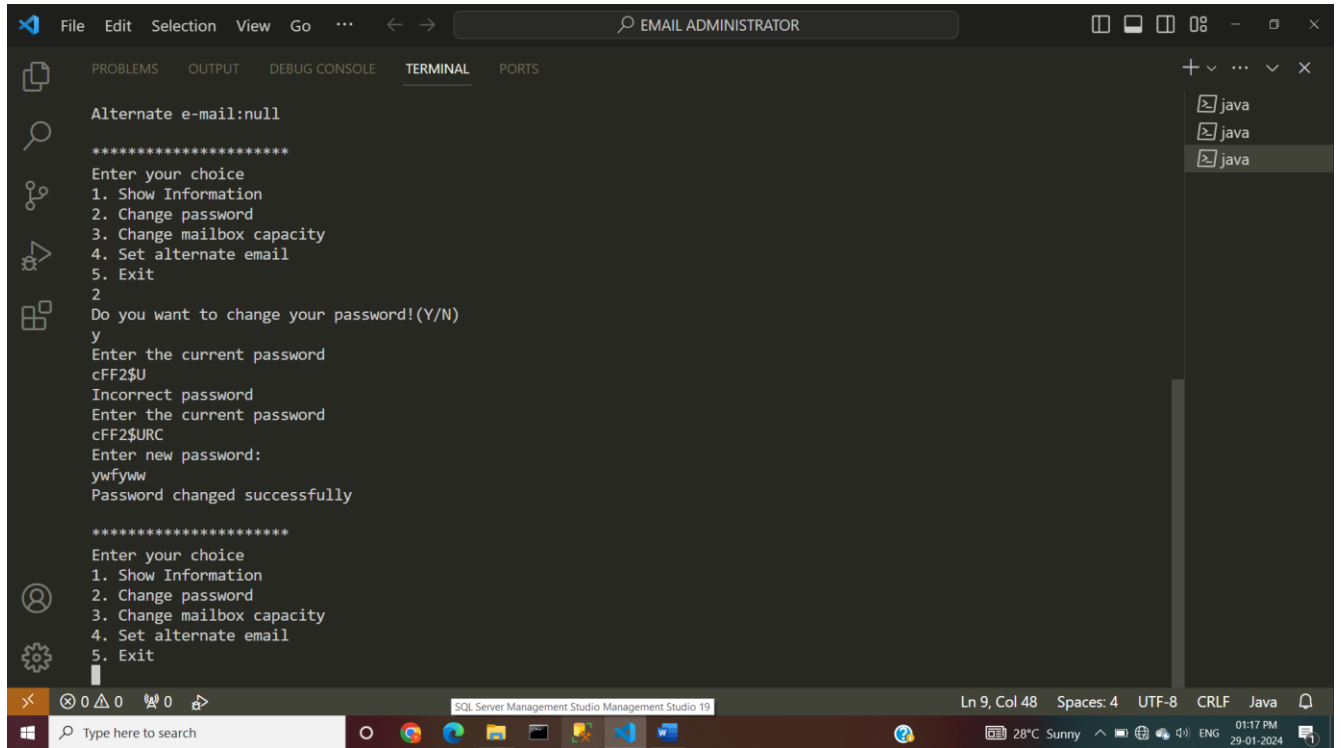
```
2
Do you want to change your password!(Y/N)
Y
Enter the current password
IzrduT8r
Enter new password:
Yashu
Password changed successfully

*****
Enter your choice
1. Show Information
2. Change password
3. Change mailbox capacity
4. Set alternate email
5. Exit
2
Do you want to change your password!(Y/N)
n
password changed option cancelled!

*****
Enter your choice
1. Show Information
2. Change password
3. Change mailbox capacity
4. Set alternate email
5. Exit
```

# Email Administrator

OUTPUT 3: changing password (incorrect password)

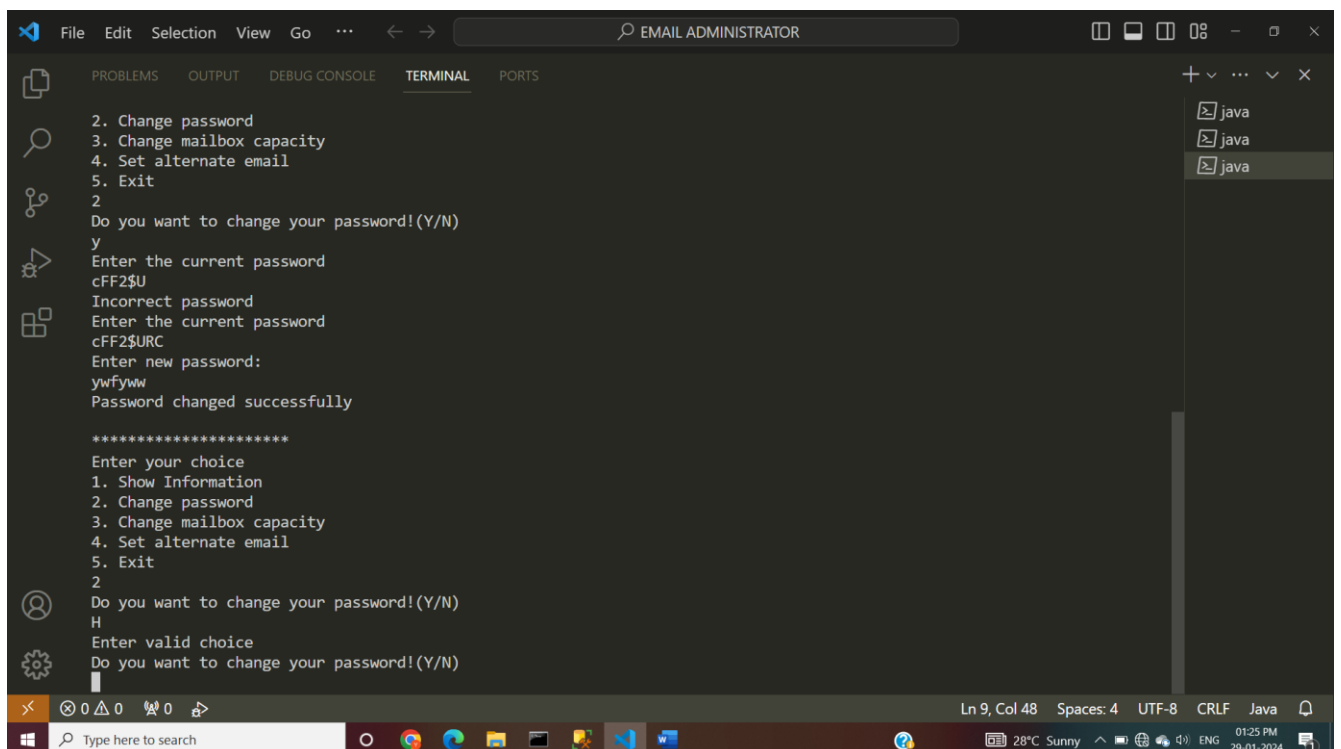


```
File Edit Selection View Go ... EMAIL ADMINISTRATOR
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Alternate e-mail:null
*****
Enter your choice
1. Show Information
2. Change password
3. Change mailbox capacity
4. Set alternate email
5. Exit
2
Do you want to change your password!(Y/N)
y
Enter the current password
cFF2$U
Incorrect password
Enter the current password
cFF2$URC
Enter new password:
ywfyyw
Password changed successfully

*****
Enter your choice
1. Show Information
2. Change password
3. Change mailbox capacity
4. Set alternate email
5. Exit

```

OUTPUT 4: Neither yes/no



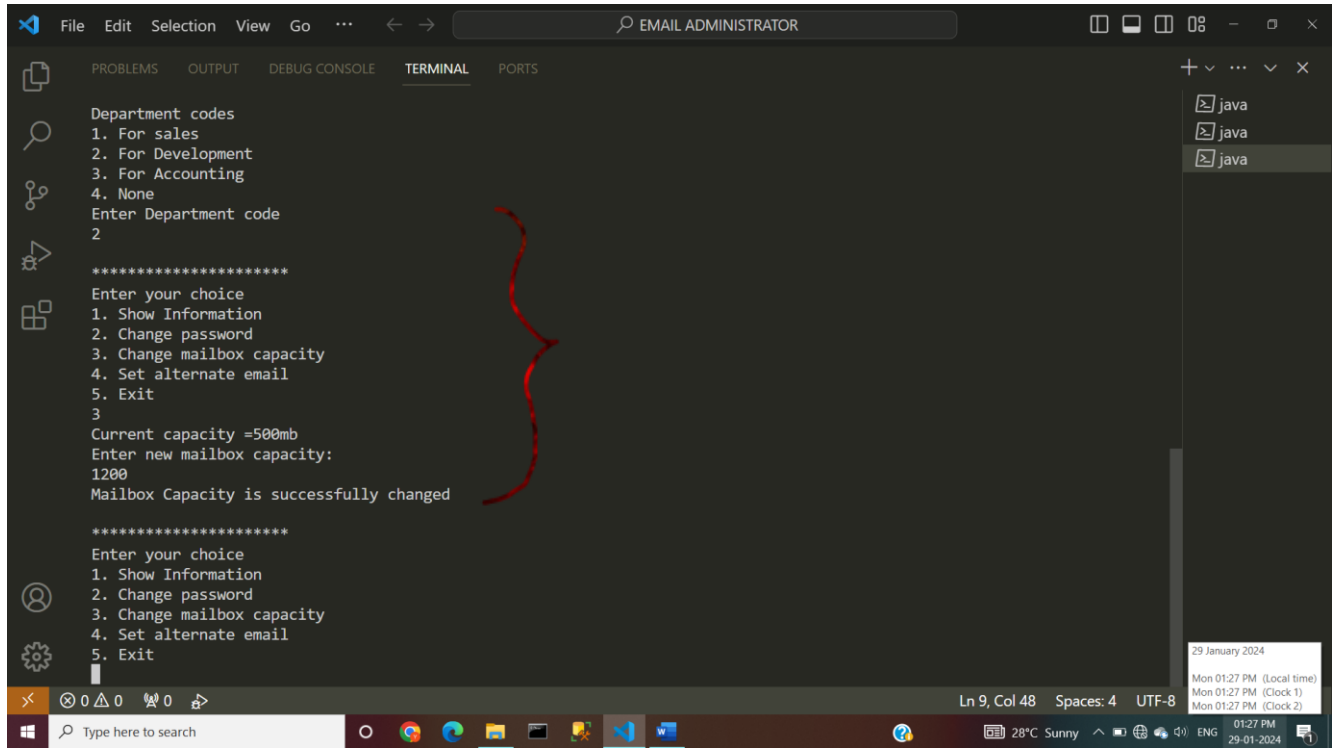
```
File Edit Selection View Go ... EMAIL ADMINISTRATOR
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
2. Change password
3. Change mailbox capacity
4. Set alternate email
5. Exit
2
Do you want to change your password!(Y/N)
y
Enter the current password
cFF2$U
Incorrect password
Enter the current password
cFF2$URC
Enter new password:
ywfyyw
Password changed successfully

*****
Enter your choice
1. Show Information
2. Change password
3. Change mailbox capacity
4. Set alternate email
5. Exit
2
Do you want to change your password!(Y/N)
H
Enter valid choice
Do you want to change your password!(Y/N)

```

# Email Administrator

## OUTPUT 5 : Changing Mailbox Capacity

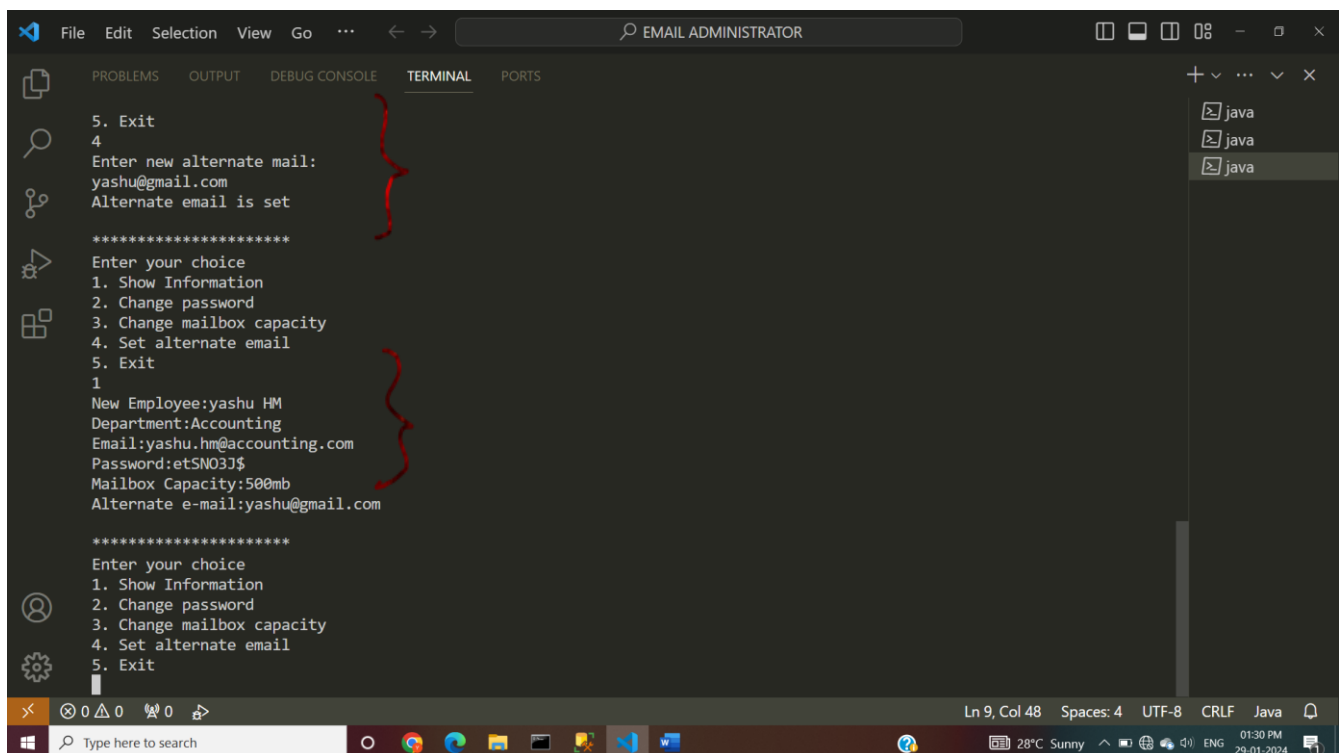


The screenshot shows the 'EMAIL ADMINISTRATOR' application running in a terminal. The interface includes a menu on the left with icons for file operations, search, and settings. The terminal output shows the following sequence of events:

- Department codes are listed: 1. For sales, 2. For Development, 3. For Accounting, 4. None.
- The user enters '2' for the Department code.
- A menu is displayed: 1. Show Information, 2. Change password, 3. Change mailbox capacity, 4. Set alternate email, 5. Exit.
- The user enters '3' to change mailbox capacity.
- The current capacity is shown as 500mb.
- The user enters '1200' for the new mailbox capacity.
- A message confirms: 'Mailbox Capacity is successfully changed'.
- The menu is shown again, and the user enters '5' to exit.

Hand-drawn red curly braces group the menu options and the user's choices. The terminal status bar at the bottom shows 'Ln 9, Col 48 Spaces: 4 UTF-8'.

## OUTPUT 6: Alternate mail set and viewed



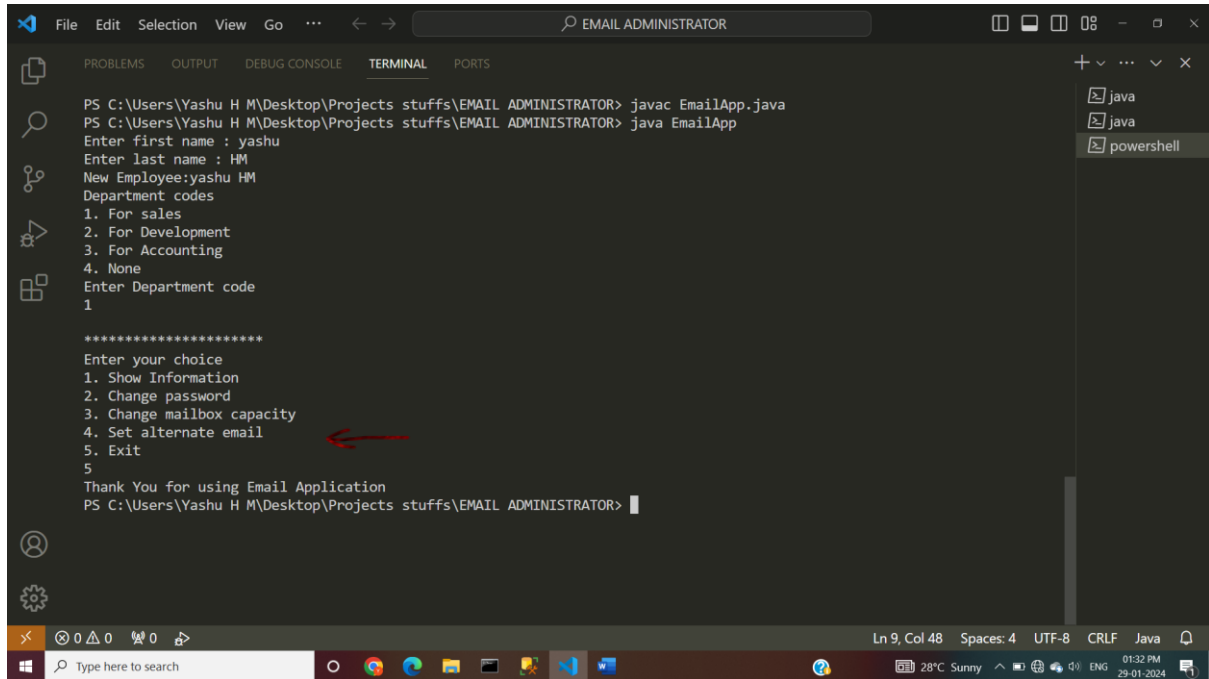
The screenshot shows the 'EMAIL ADMINISTRATOR' application running in a terminal. The interface is similar to the previous one. The terminal output shows the following sequence of events:

- The user enters '5' to exit the previous menu.
- The user enters '4' to set an alternate email.
- The user enters 'yashu@gmail.com' for the new alternate mail.
- A message confirms: 'Alternate email is set'.
- The menu is displayed again.
- The user enters '1' to show information.
- The employee details are displayed: New Employee:yashu HM, Department:Accounting, Email:yashu.hm@accounting.com, Password:etSN03J\$, Mailbox Capacity:500mb, Alternate e-mail:yashu@gmail.com.
- The menu is shown again, and the user enters '5' to exit.

Hand-drawn red curly braces group the menu options and the user's choices. The terminal status bar at the bottom shows 'Ln 9, Col 48 Spaces: 4 UTF-8 CRLF Java'.

# Email Administrator

## OUTPUT 7: Exiting the application



```
PS C:\Users\Yashu H M\Desktop\Projects stuffs\EMAIL ADMINISTRATOR> javac EmailApp.java
PS C:\Users\Yashu H M\Desktop\Projects stuffs\EMAIL ADMINISTRATOR> java EmailApp
Enter first name : yashu
Enter last name : HM
New Employee:yashu HM
Department codes
1. For sales
2. For Development
3. For Accounting
4. None
Enter Department code
1

*****
Enter your choice
1. Show Information
2. Change password
3. Change mailbox capacity
4. Set alternate email
5. Exit
5
Thank You for using Email Application
PS C:\Users\Yashu H M\Desktop\Projects stuffs\EMAIL ADMINISTRATOR>
```