PM SHRI KENDRIYA VIDYALAYA BERHAMPUR

SESSION 2024-25

A Project Report on

ONLINE SPORTS SOFTWARE

For

CBSE 2025 Examination

[As a part of the Computer Science Course (083)]

SUBMITTED BY :

Sidhanta Patra

Class- XII A

SUBMITTED TO:

Mr. S.K.Misra

PGT(Computer science)

THIS IS TO CERTIFY THAT SIDHANT PATRA STUDYING IN CLASS XII A, HAS SATISFACTORILY COMPLETED PROJECT WITH THE TITLE ONLINE SPORTS SOFTWARE UNDER THE GUIDANCE OF Mr. SAROJ KANTA MISRA, PGT (COMPUTER SCIENCE) DURING THE ACADEMIC YEAR 2024-25 IN PARTIAL FULFILLMENT OF "COMPUTER SCIENCE" PRACTICAL EXAMINATION OF CENTRAL BOARD OF SECONDARY EXAMINATION (CBSE)

_____
Internal Examiner

_____
External Examiner

_____
PRINCIPAL

ACKNOWLEDGEMENT

I warmly acknowledge the continuous encouragement and timely suggestions offered by our Principal Mr. SHIVAPRIYA DASH. I extend my hearty thanks for giving me the opportunity to make use of the facilities available in the campus to carry out the project successfully.

I am highly indebted to Mr. SAROJ KANTA MISRA, (PGT Computer Science), for the constant supervision, providing necessary information and supporting in completing the project. I would like to express my gratitude towards them for their kind cooperation and encouragement.

Finally, I extend my gratefulness to one and all who are directly or indirectly involved in the successful completion of this project work.

Name:Sidhanta patra

Sign:

Class XII-A

- Python is a high-level language. It is a free and open-source language.
- It is an interpreted language, as Python programs are executed by an interpreter.
- Python programs are easy to understand as they have a clearly defined syntax and relatively simple structure.
- Python is case-sensitive. For example, NUMBER and number are not same in Python.
- Python is portable and platform independent, means it can run on various operating systems and hardware platforms.
- Python has a rich library of predefined functions.
- Python is also helpful in web development. Many popular web services and applications are built using Python.
- Python uses indentation for blocks and nested blocks.

CONTENTS

System Implementation

Hardware used:

While developing the software, the used hardware's are: PC with Intel Core i3 processor having 4.00 GB RAM and other required devices.

Software used:

- Microsoft Windows® 10 as Operating System.
- Python IDLE as Front-end Development environment.
- CSV Files as Back-end for storing the data.

MS-Word 2010 for documentation

INTRODUCTION

AN ONLINE SPORTS SOFTWARE

(XTREME SPORTS)

Welcome to XTreme Sports, your one-stop shop for all your extreme sports needs! Whether you're a seasoned pro or just starting out, we have everything you need to get your adrenaline pumping.

Our user-friendly website makes it easy to browse our extensive inventory of sports equipment, clothing, and accessories. With just a few clicks, you can find exactly what you're looking for and have it delivered right to your doorstep.

We're committed to providing our customers with the best possible shopping experience. That's why we offer a wide range of 1 payment options and a hassle-free return policy

PYTHON SOURCE CODE

```
import random
import mysql.connector as m

from tabulate import tabulate as ta



db=m.connect(host="localhost",user="root",password="admin",database="xtreme")

if db.is_connected():

   print("establishment successful")
else:
   print("error")
co=db.cursor()




def fo():
   print('''___WELCOME TO XTREME SPORTZ___
        HERE YOU WILL GET YOUR FAVOURITE SPORTS PRODUCTS
        SO PLAY WITH A STYLE''')

   try:
      print("enter a number to prove you are not a robot")
      cho=int(input("enter the num "))
```

```
        custu()
    except:
        print(" OOPS! ENTER A NUMBER ONLY")




# custu()









#FOR ADMINISTRATOR
def ad_order():
    lo=int(input('''ENTER THE FOLOWING CHOICES
                    ENTER 1 FOR ADDING ITEM
                    ENTER 2 FOR DELETING ITEM
                    ENTER 3 FOR RETURNING TO MAIN MENU
                    ENTER 4 FOR VIEWING ALL ITEM'''))

    if lo==1:
        additem()
    elif lo==2:
        delitem()
    elif lo==3:
        custu()
    elif lo==4:
        view_all_item()
    elif lo==5:
        p_order()
```

```
def additem():
    # import mysql.connector as m
    db=m.connect(host="localhost",user="root",password="admin",database="xtreme")
    co=db.cursor()

    pid=int(input("enter the product id"))
    pname=input("enter the product name")
    company=input("enter the company that produces product")
    sport=input("enter the sport")
    price=int(input("enter the price"))
    size=input("enter the size")
    co.execute("insert into product values({},'{}',{},'{}','{}',{})".format(pid,pname,company,sport,size))
    db.commit()
    nh=input(''' IF YOU WANT TO CONTINUE
                TYPE YES
                IF YOU DO NOT WANT
                TYPE NO''')
    if nh.lower()=="yes":
        additem()
    else:

        ad_order()



def delitem():
    # import mysql.connector as m
    db=m.connect(host="localhost",user="root",password="admin",database="xtreme")
    co=db.cursor()
```

```
    p_id=int(input("enter the product you want to delete"))
    co.execute("delete from product where pid='{}'".format(p_id))
    db.commit()
    ph=input("if you want to continue-----type yes/no")
    if ph=="yes":
        delitem()

    else:
        ad_order()

def view_all_item():
    # import mysql.connector as m
    ''' display all products available in store'''
    db=m.connect(host="localhost",user="root",password="admin",database="xtreme")
    co=db.cursor()
    co.execute("select * from product")
    vo=co.fetchall()
    c=0
    print(ta(vo,headers=["pid","Pname ","Company Name","Sports" ,"Price","Size"],tablefmt="psql"))

    print(''' enter 1 to go into administrator order menu
    enter 2 for going to order menu''')
    nh=int(input("enter the choice: "))
    if nh==1:
        ad_order()
    if nh==2:
        order()
    else:
        print(" invalid choice")

#function to search for an product

def search_item():
```

```
# Database connection and cursor initialization
db = m.connect(host="localhost", user="root", password="admin", database="xtreme")
co = db.cursor()

try:
    # Ask the user for the product name
    pname = input("Enter the product name: ")
    co.execute("SELECT * FROM product WHERE pname = '{}'".format(pname))
    go = co.fetchall()

    # If product(s) found, display them
    if go:
        for i in go:
            print(i)
    else:
        print("No product found with that name.")

except:
    print("Invalid product name.")
    search_item()  # Recursively call the function if there's an error

try:
    # Ask the user for the company name (for further filtering)
    company = input("Enter the company name of the product to search: ")
    co.execute("SELECT * FROM product WHERE pname = '{}' AND company = '{}'".format(pname, company))
    go1 = co.fetchall()

    # If matching company found, display them
    if go1:
        for j in go1:
            print("*****")
            print(j)
            continue
```

```
    else:
        print("No matching company found for this product.")

    except:
        print("Invalid company.")
        search_item()  # Recursively call the function if there's an error

    print('''Choose size from:
            LARGE
            MEDIUM
            SMALL
            REGULAR''')

    try:
        # Define available sizes
        s = ["large", "medium", "small", "regular"]
        size = input("Enter the size of the product to search: ")

        if size.lower() in s:
            # Search for the product by name, company, and size
            co.execute("SELECT * FROM product WHERE pname = '{}' AND company = '{}' AND size = '{}'".format(pname,
company, size))
            re = co.fetchall()

            # If a matching product is found, display it
            if re:
                print("*****")
                print(re)
            else:
                print("No matching product found for the selected size.")
        else:
            print("Invalid size entered. Please choose from the available options.")

    except:
```

```
        print("Please enter a valid size.")

#handle all the orders by the order function
#we can place order for any selected product of choice

def p_order():



    db=m.connect(host="localhost",user="root",password="admin",database="xtreme")
    co=db.cursor()
    # Establish a database connection
    db = m.connect(host="localhost", user="root", password="admin", database="xtreme")
    co = db.cursor()

    print("------ Place Your Order ------")
    try:
        cust_id=int(input("enter customer id"))
        product_id = int(input("Enter the Product ID: "))
        customer_name = input("Enter your name: ")
        address = input("Enter your delivery address: ")
        quantity = int(input("Enter the quantity: "))



        co.execute("SELECT pname, price, size, company FROM product WHERE pid = %s", (product_id,))
        product = co.fetchone()

        if product:
            product_name, price_per_unit, size, company = product
            total_price = quantity * price_per_unit

            # order details for confirmation
            print("\n------ Order Details ------")
            print(f"Product Name: {product_name}")
```

```
print(f"Company: {company}")
print(f"Price per Unit: {price_per_unit}")
print(f"Selected Quantity: {quantity}")
print(f"Total Price: {total_price}")
print(f"Available Size: {size}")

# Confirm the size
selected_size = input("Enter your size: ")
if selected_size != size:
    print("Selected size is not available. Please try again.")
    return

# Confirm the order
confirm = input("Do you want to confirm this order? (yes/no): ").lower()
if confirm == "yes":
    #  a randomly delivery ID
    delivery_id = random.randint(100000, 999999)

    # Insert order into the delivery table
    co.execute(
        "INSERT INTO delivery (delivery_id,c_id,customer_name, product_name, company, total_price,address) "
        "VALUES (%s, %s ,%s, %s, %s, %s, %s)",
        (delivery_id, cust_id , customer_name, product_name, company, total_price,address))

    db.commit()

    # Step 5: Generate a bill if requested
    print("\n*** YOUR ORDER IS PLACED SUCCESSFULLY ***")
    generate_bill = input("Do you want the bill? (yes/no): ").lower()
    if generate_bill == "yes":
        print("\n------ Bill ------")
        print(f"Product ID: {product_id}")
```

```
            print(f"Product Name: {product_name}")
            print(f"Company: {company}")
            print(f"Price per Unit: {price_per_unit}")
            print(f"Quantity: {quantity}")
            print(f"Total Price: {total_price}")
            print(f"Delivery Address: {address}")
            print(f"Delivery ID: {delivery_id}")
        else:
            print("Order canceled.")
    else:
        print("Product not found. Please try again.")
except Exception as e:
    print(f"An error occurred: {e}")
finally:
    db.close()

print(input("any key to continue: "))

#for canceing order
def p_cancel():
    # import mysql.connector as m
    db=m.connect(host="localhost",user="root",password="admin",database="xtreme")
    co=db.cursor()
    try:
        print("Processing...")
        print("opening yours order list...")
        #display all orders
        co.execute("select * from delivery")
        orders=co.fetchall()
        if not orders:
            print("no orders found! ")
            return
        #displaying all orders in tabulated form
```

```
    print(ta(orders,headers=["delivery ID","customer id","customer name","product
name","company","price","date","address"],tablefmt="grid"))
    order_id=int(input("enter the id you want to cancel"))

    co.execute("select * from deliver where order_id=%s",(order_id,))
    order=db.fetchone()
    if order:
      confirm=input("ARE YOU SURE YOU WANT TO CANCEL THIS ORDER ?(yes/no) : ").lower()
      if confirm=="yes":
        co.execute("delete from delivery where order_id= %s ", (order_id,))

        db.commit()
        print("Order successfully canceled")
      else:

        print("cancellation aborted")
    else:
      print("invalid order id ! no such id found")
  except Exception as e:
    print(f"An error occurred : {e} ")
  finally:

    print("_____________")


    print("enter 1- if you want to continue canceling another orders")
    print("enter 2- if you want to check your orders")
    print("enter 3- if you want to go for order menu")
    print("_____________")

  enter=int(input("Enter Number: " ))
  if enter==1:
    p_cancel()
```

```
    elif enter==2:
        co.execute("select * from delivery")
        orders=co.fetchall()
        print(ta(orders,headers=["delivery ID","customer id","customer name","product
name","company","price","date","address"],tablefmt="grid"))


    elif enter==3:
        order()
    else:
        print("invalid choice!")







    print(input('''press any key to continue'''))




def return_deli():
    db = m.connect(host="localhost", user="root", password="admin", database="xtreme")
    co = db.cursor()
    print("------ Return a Product ------")
    try:
        delivery_id = int(input("Enter your Delivery ID: "))
        reason = input("Enter the reason for returning the product: ")
        co.execute("SELECT customer_name, product_name, total_price, address FROM delivery WHERE delivery_id =
%s", (delivery_id,))
        delivery = co.fetchone()
        if delivery:
            customer_name, product_name, total_price, address = delivery
            # Display return details for confirmation
            print("\n------ Return Details ------")
            print(f"Customer Name: {customer_name}")
            print(f"Product Name: {product_name}")
```

```
            print(f"Total Price: {total_price}")
            print(f"Delivery Address: {address}")
            print(f"Reason for Return: {reason}")
            # Step 3: Confirm the return
            confirm = input("Do you want to confirm this return? (yes/no): ").lower()
            if confirm == "yes":
                # Insert return details into the returns table
                co.execute(
                    "INSERT INTO return (delivery_id, customer_name, product_name, total_price, address, reason) "
                    "VALUES (%s, %s, %s, %s, %s, %s)",
                    (delivery_id, customer_name, product_name, total_price, address, reason)
                )
                db.commit()

                # Step 4: Notify successful return
                print("\n** YOUR RETURN HAS BEEN PROCESSED SUCCESSFULLY **")
            else:
                print("Return canceled.")
        else:
            print("Invalid Delivery ID. Please try again.")
    except Exception as e:
        print(f"An error occurred: {e}")

def show_deliveryhist():

    db=m.connect(host="localhost",user="root",password="admin",database="xtreme")
    co=db.cursor()
    print("_____")

    print("product you have order till now")
    try:
        cust_id=int(input("enter the customer id  : "))
        co.execute("select * from delivery where c_id ={} ".format(cust_id))
```

```
        f=co.fetchall()
        if f:
            print(ta(f,headers=["delivery ID ","c_id", " cust_name", " product
name","company","price","address"],tablefmt="grid"))
        else:
            print("no such orders found")
    except Exception as e:
        print(f"An error occurred: {e}")



    print("--------------------------------")


def p_history():
    db=m.connect(host="localhost",user="root",password="admin",database="xtreme")
    co=db.cursor()
    #to show all the purchases


    try:
        co.execute("select * from delivery")
        p=co.fetchall()
        if p:
            print(ta(p,headers=["delivery ID ","c_id", " cust_name", " product
name","company","price","address"],tablefmt="grid"))
        else:
            print("no purchase history found")
    except Exception as e:
        print(f"An error occured : {e}")


#DELIVERY MENU
def delivery():
    print("DELIVERY MENU")


    print('''ENTER THE FOLOWING CHOICES
                    ENTER 1 FOR PLACING ORDER
```

```
                ENTER 2 FOR CANCELLING ITEM
                ENTER 3 FOR RETURNING ORDER
                ENTER 4 FOR ORDER MENU''')
    fi=int(input("enter the choices: "))
    if fi==1:
        p_order()
    elif fi==2:
        p_cancel
    elif fi==3:
        return_deli()
    elif fi==4:
        order()
    else:
        print("invalid options")
        delivery()



#FOR ADMINISTRATOR
def ad_order():
    print('''ENTER THE FOLOWING CHOICES
                ENTER 1 FOR ADDING ITEM
                ENTER 2 FOR DELETING ITEM
                ENTER 3 VIEW ALL ITEMS
                ENTER 4 FOR VIEWING PURCHASE HISTORY
                ENTER 5 FOR RETURNING TO ORDER MENU
                ENTER 6 FOR MAIN MENU''')
    try:
        lo=int(input("enter your choice :- "))


        if lo==1:
```

```
        additem()
    elif lo==2:
        delitem()
    elif lo==3:
        view_all_item()
    elif lo==4:
        p_history()
    elif lo==5:
        order()
    elif lo==6:
        custu()


  except:
    print("invalid option")
    ad_order()


#FOR USER
def order():
  while True:
    print('''  WELOME TO ORDER MENU ''')
    print('''ENTER THE FOLOWING CHOICES
                ENTER 1 FOR VIEWING ALL ITEM
                ENTER 2 FOR SEARCHING ITEM
                ENTER 3 TO VIEW  CURRENT DELIVERY HISTORY
                ENTER 4 FOR MAIN MENU
                ENTER 5 TO GO TO DELIVERY MENU
                ENTER 6 FOR EXIT ''')
    try:
      ko=int(input("enter your choice :- "))
      if ko==1:
        view_all_item()
```

```
        elif ko==2:
            search_item()
        elif ko==3:
            show_deliveryhist()
        elif ko==4:
            custu()
        elif ko==5:
            delivery()
        elif ko==6:
            print("thank you")
            break

    except:
        print("invalid option")
        order()



def sign_up():
    # import mysql.connector as m
    db=m.connect(host="localhost",user="root",password="admin",database="xtreme")
    co=db.cursor()
    try:
        cus_id=int(input("enter the customer id :- "))

        name=input("enter your username:- ")
        co.execute("select count(*) from customer where name='{}'".format(name))
        o=co.fetchone()
        if o[0]>0:
            print("username already taken")
            sign_up()
        else:
```

```
f=input("enter your email:-  ")
if f[0]!="@" and "." in f and f.count("@")==1 and f[-1]!=".":
    s = f.split(".")[-1]
    for i in s:
        if not i.isdigit():
            pass
        else:
            print("Enter valid Email!!!!")
            return
address=input("enter your address")
while True:
    ph=input("enter your phone number: ")
    if len(ph)==10:
        print("is this your correct number or you want to change it? type yes or no ")
        descision=input("enter your choice")
        if descision.lower()=="no":
            break
    else:
        print("OOPS INVALID NUMBER")
c=3
flag=1
while flag:
    otp=random.randrange(100000,999999)
    while c>0:
        print(otp)
        print("we have sent you a 6-digit code on your registered mobile name ")
        otp_c=int(input("enter the given otp: "))
        if otp_c==otp:
            print("you have registered successfully")
            co.execute("insert into customer values({},'{}','{}','{}',{})".format(cus_id,name,f,address,ph)    )



            flag=flag-1
```

```
                order()
                db.commit()
                break
            else:
                print("INVALID OTP TRY AGAIN")
                c=c-1
                if c==0:
                    c=3
                    print("we will resend you a new one")
                    break
    except Exception as e:
        print(f"An error occured: {e}")




#login
def log():
    # import mysql.connector as m

    db=m.connect(host="localhost",user="root",password="admin",database="xtreme")
    co=db.cursor()
    try:
        w2=input("ENTER YOUR REGITERED NAME  :- ")
        ci=int(input("enter the customer id: - "))
        co.execute("select name,cust_id from customer where name='{}' and cust_id ={}".format(w2,ci))
        b=co.fetchall()
        if b is not None:
            print('''LOGIN SUCCESSFUL
                WELCOME BACK''')
            order()
            db.commit()
```

```
        else:
            print("sorry we can not find your registered name")
    except Exception as e:
        print(f"An error occurre:  {e}")


    print("**********")
def admin_log():


    ps=input("enter the password:- ")
    if ps=="romanreigns":
        print("password granted")
        ad_order()
    else:
        print("wrong password")
        custu()
print("**********")


#delete customer
def delcustdet():
    db=m.connect(host="localhost",user="root",password="admin",database="xtreme")
    co=db.cursor()
    try:


        cust_name=input("enter the customer name you want to delete")
        co.execute("select * from customer where name='{}'".format(cust_name))
        cp=co.fetchone()
        if cp:
            co.execute("delete from product where pname='{}'".format(cust_name))
            db.commit()
        else:
            print("USERNAME NOT FOUND")
            delcustdet()
```

```
    nh=input(''' IF YOU WANT TO CONTINUE
                TYPE 1
                IF YOU DO NOT WANT
                TYPE 2
                TO EXIT TYPE ANYTHING''')




    if int(nh)==1:
       delcustdet()
    elif int(nh)==2:
       updatecus()
    else:
       print("thank you")

  except Exception as e:
    print(f"An error occured: {e}")
    updatecus()



#update customer
def updatenamefromcus():
  # import mysql.connector as m
  db=m.connect(host="localhost",user="root",password="admin",database="xtreme")
  co=db.cursor()
  try:


    old_name=input("enter the customer name you want to update : ")
    cust_name=input("enter the new customer name : ")
    if old_name==cust_name:
       print("username already taken")
       updatenamefromcus()
    else:
```

```
        co.execute("select * from customer where name='{}'".format(old_name))
        cp=co.fetchall()
        if cp:
            co.execute("update customer set name='{}' where name='{}'".format(cust_name,old_name))
            db.commit()
            print("updated successfully")
        else:
            print("name not found")
            updatecus()
        print(''' IF YOU WANT TO CONTINUE
                TYPE 1
                IF YOU DO NOT WANT
                TYPE 2

                ENTER ANYTHING TO STOP''')
        nh=input("enter choice")

        if int(nh)==1:
            updatenamefromcus()
        elif int(nh)==2:
            updatecus()
        else:
            print("thank you")

    except Exception as e:
        print(f"An error occured: {e}")
        updatenamefromcus()



def updateaddress():
    db=m.connect(host="localhost",user="root",password="admin",database="xtreme")
    co=db.cursor()
```

```
try:
    new_add=input("enter new address ")
    cust_name=input("enter the  customer name")
    co.execute("select count(*) from customer where name='{}'".format(cust_name))
    cp=co.fetchone()
    if cp[0]==0:
        print("no usernamefound")
        updatecus()
    else:
        co.execute("update customer set address='{}' where name='{}'".format(new_add,cust_name))
        print("updated successfully")
        db.commit()


    print(''' IF YOU WANT TO CONTINUE
            TYPE 1
            IF YOU DO NOT WANT
            TYPE 2
            ENTER ANYTHING TO STOP''')
    nh=input("enter choice :")

    if int(nh)==1:
        updateaddress()
    elif int(nh)==2:
        updatecus()
    else:
        print("thank you")


except Exception as e:
    print(f"An error occured: {e}")
    updatecus()
```

```
def updatephonenum():
    db=m.connect(host="localhost",user="root",password="admin",database="xtreme")
    co=db.cursor()
    try:
        cus_id=input("enter the customer you want to update")
        ph_no=int(input("enter the phone number"))



        co.execute("select count(*) from customer where name='{}'".format(cus_id))
        cp=co.fetchone()
        if cp[0]==0:
            print("no usernamefound")
            updatecus()
        else:

            co.execute("update customer set phone={} where name='{}'".format(ph_no,cus_id))



            db.commit()
        print("phone number updated successfully")
        print(''' IF YOU WANT TO CONTINUE
                TYPE 1
                IF YOU DO NOT WANT
                TYPE 2
                ENTER ANYTHING TO STOP''')
        nh=input("enter choice: ")

        if int(nh)==1:
            updatecus()
        elif int(nh)==2:
            updatephonenum()
```

```
        else:
            print("thank you")


    except Exception as e:
        print(f"An error occured: {e}")
        updatephonenum()



def updatecus():
    print('''ENTER THE FOLOWING CHOICES

                        ENTER 1 FOR UPDATING NAME FROM CUSTOMER
                        ENTER 2 FOR UPDATING ADDRESS
                        ENTER 3 FOR UPDATING PHONE NUMBER
                        ENTER 4 FOR THANK YOU
                      ''')
    try:
        f=int(input("enter your choice: "))
        if f==1:
            updatenamefromcus()
        if f==2:
            updateaddress()
        if f==3:
            updatephonenum()
        if f==4:
            print("thank you")
            custu()
    except:
        print("invalid choice")

    print("--------------------------------")
```

```
def custu():


    db=m.connect(host="localhost",user="root",password="admin",database="xtreme")
    co=db.cursor()
    while True:
        print('''MAIN_MENU
                ENTER 0 FOR EXIT
                ENTER 1 FOR SIGN UP
                ENTER 2 FOR ADMIN LOGIN
                ENTER 3 TO DELETE CUSTOMER DETAILS
                ENTER 4 UPDATE CUSTOMER
                ENTER 5 FOR USER LOGIN
                  ''')
        try:
            cho=int(input("enter your choice"))
            if cho==1:
                sign_up()
            elif cho==2:
                admin_log()
            elif cho==5:
                log()
            elif cho==3:
                delcustdet()
            elif cho==4:
                updatecus()
            elif cho==0:
                print("Thank You For Visiting Our Shop!!!!")
                break


        except:
            print("invalid choice")
fo()
```

- Program outputs

Main menu

```
establishment successful
*****************************
______WELCOME TO XTREME SPORTZ______
        HERE YOU WILL GET YOUR FAVOURITE SPORTS PRODUCTS
        SO PLAY WITH A STYLE
enter a number to prove you are not a robot
enter the num 2
MAIN_MENU
                        ENTER 0 FOR EXIT
                        ENTER 1 FOR SIGN UP
                        ENTER 2 FOR ADMIN LOGIN
                        ENTER 3 TO DELETE CUSTOMER DETAILS
                        ENTER 4 UPDATE CUSTOMER
                        ENTER 5 FOR USER LOGIN

enter your choice2
```

Admin Menu

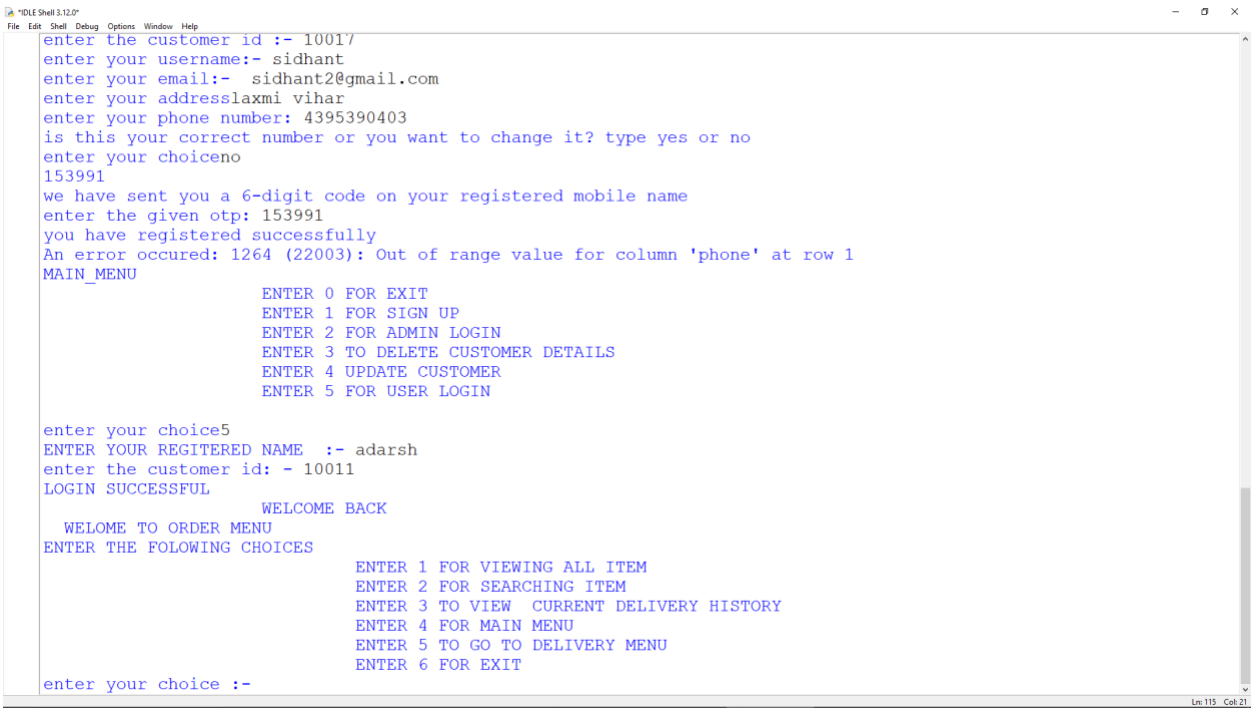
```
enter your choice2
enter the password:- romanreigns
password granted
ENTER THE FOLOWING CHOICES
                            ENTER 1 FOR ADDING ITEM
                            ENTER 2 FOR DELETING ITEM
                            ENTER 3 VIEW ALL ITEMS
                            ENTER 4 FOR VIEWING PURCHASE HISTORY
                            ENTER 5 FOR RETURNING TO ORDER MENU
                            ENTER 6 FOR MAIN MENU
enter your choice :- 4
+----------------+--------+--------------+-----------------+-----------+---------+---------------+
|   delivery ID  |   c_id |  cust_name   |  product name   | company   |   price | address       |
+================+========+==============+=================+===========+=========+===============+
|         123456 |  10011 | adarsh       | bat             | mrf       |  500000 | ambapua       |
+----------------+--------+--------------+-----------------+-----------+---------+---------------+
|         123457 |  10012 | ajay         | bat             | mrf       |   56000 | ambapua       |
+----------------+--------+--------------+-----------------+-----------+---------+---------------+
|         123458 |  10013 | anirudh      | ball            | club      |    2000 | old bus stand |
+----------------+--------+--------------+-----------------+-----------+---------+---------------+
|         123459 |  10014 | virat        | football        | nike      |    2000 | old bus stand |
+----------------+--------+--------------+-----------------+-----------+---------+---------------+
|         123460 |  10015 | aayuosh      | football        | adidas    |    3000 | tulsi vihar   |
+----------------+--------+--------------+-----------------+-----------+---------+---------------+
|         123461 |  10016 | sujal        | badminton       | yonex     |    1500 | tulsi vihar   |
+----------------+--------+--------------+-----------------+-----------+---------+---------------+
|         123462 |  10017 | sankit       | badminton       | yonex     |    2000 | tata bench    |
```

Sign up

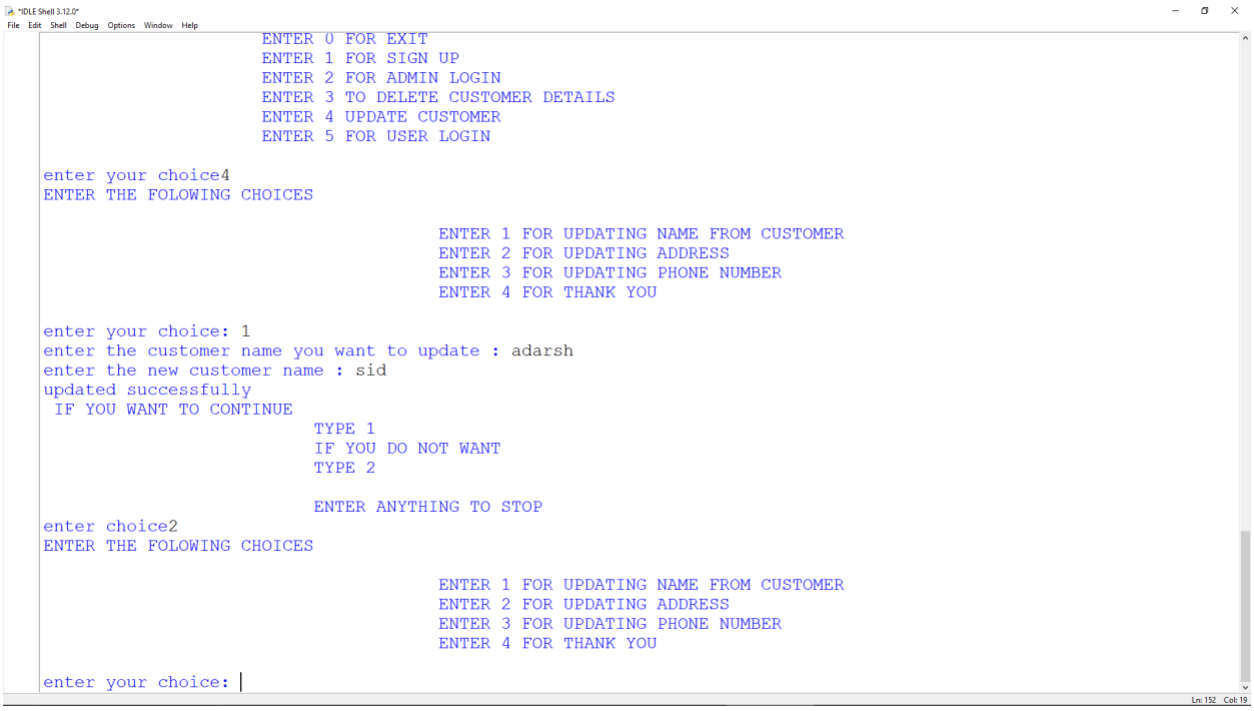
```
                    ENTER 0 FOR EXIT
                    ENTER 1 FOR SIGN UP
                    ENTER 2 FOR ADMIN LOGIN
                    ENTER 3 TO DELETE CUSTOMER DETAILS
                    ENTER 4 UPDATE CUSTOMER
                    ENTER 5 FOR USER LOGIN

enter your choice1
enter the customer id :- 10017
enter your username:- sidhant
enter your email:-  sidhant2@gmail.com
enter your addresslaxmi vihar
enter your phone number: 4395390403
is this your correct number or you want to change it? type yes or no
enter your choiceno
153991
we have sent you a 6-digit code on your registered mobile name
enter the given otp: 153991
you have registered successfully
```

Log in



Update table



Order menu

```
  WELOME TO ORDER MENU
ENTER THE FOLOWING CHOICES
                              ENTER 1 FOR VIEWING ALL ITEM
                              ENTER 2 FOR SEARCHING ITEM
                              ENTER 3 TO VIEW  CURRENT DELIVERY HISTORY
                              ENTER 4 FOR MAIN MENU
                              ENTER 5 TO GO TO DELIVERY MENU
                              ENTER 6 FOR EXIT
```

Search item

```
enter your choice :- 2
Enter the product name: handball
(202, 'handball', 'nike', 'handball', 600, 'regular')
(203, 'handball', 'adidas', 'handball', 700, 'regular')
(204, 'handball', 'nivea', 'handball', 1000, 'regular')
Enter the company name of the product to search: nike
***************
(202, 'handball', 'nike', 'handball', 600, 'regular')
Choose size from:
                 LARGE
                 MEDIUM
                 SMALL
                 REGULAR
Enter the size of the product to search: regular
***************
[(202, 'handball', 'nike', 'handball', 600, 'regular')]
```

Delivery history of an user

```
ENTER THE FOLOWING CHOICES
                            ENTER 1 FOR VIEWING ALL ITEM
                            ENTER 2 FOR SEARCHING ITEM
                            ENTER 3 TO VIEW  CURRENT DELIVERY HISTORY
                            ENTER 4 FOR MAIN MENU
                            ENTER 5 TO GO TO DELIVERY MENU
                            ENTER 6 FOR EXIT
enter your choice :- 3
_____________________________
product you have order till now
enter the customer id  : 10011
+---------------+--------+-------------+----------------+-----------+---------+-----------+
|   delivery ID  |   c_id |  cust_name   |  product name   | company   |   price | address   |
+================+========+=============+================+===========+=========+===========+
|         123456 |  10011 | adarsh       | bat             | mrf       |  500000 | ambapua   |
+----------------+--------+-------------+----------------+-----------+---------+-----------+
--------------------------------
```

Delivery menu

```
enter your choice :- 5
DELIVERY MENU
ENTER THE FOLOWING CHOICES
                                    ENTER 1 FOR PLACING ORDER
                                    ENTER 2 FOR CANCELLING ITEM
                                    ENTER 3 FOR RETURNING ORDER
                                    ENTER 4 FOR ORDER MENU
enter the choices:
```

Placing order

```
enter the choices: 1
------ Place Your Order ------
enter customer id10011
Enter the Product ID: 200
Enter your name: adarsh
Enter your delivery address: gc vihar
Enter the quantity: 3

------ Order Details ------
Product Name: hockeystick
Company: adidas
Price per Unit: 500
Selected Quantity: 3
Total Price: 1500
Available Size: regular
Enter your size: regular
Do you want to confirm this order? (yes/no): yes

******* YOUR ORDER IS PLACED SUCCESSFULLY *******
Do you want the bill? (yes/no): yes
```

Returning an order

```
enter the choices: 3
------ Return a Product ------
Enter your Delivery ID: 104794
Enter the reason for returning the product: bekar hain

------ Return Details ------
Customer Name: adarsh
Product Name: hockeystick
Total Price: 1500
Delivery Address: gc vihar
Reason for Return: bekar hain
Do you want to confirm this return? (yes/no): yes
```

Thank you

```
MAIN_MENU
                    ENTER 0 FOR EXIT
                    ENTER 1 FOR SIGN UP
                    ENTER 2 FOR ADMIN LOGIN
                    ENTER 3 TO DELETE CUSTOMER DETAILS
                    ENTER 4 UPDATE CUSTOMER
                    ENTER 5 FOR USER LOGIN

enter your choice0
Thank You For Visiting Our Shop!!!!
```

Database

Customer table

```
mysql> use xtreme;
Database changed
mysql> select * from customer;
+---------+---------+--------------------+---------------+----------+
| cust_id | name    | email              | address       | phone    |
+---------+---------+--------------------+---------------+----------+
|   10011 | sid     | adarsh23@gmail.com | gc vihar      | 72878732 |
|   10013 | anirudh | anirudh@gmail.com  | bima nagar    |       67 |
|   10014 | virat   | virat@gmail.com    | old bus stand |       68 |
|   10015 | aayuosh | virat@gmail.com    | tulsi vihar   |       70 |
+---------+---------+--------------------+---------------+----------+
4 rows in set (0.00 sec)
```

Delivery table

```
mysql> select * from delivery;
+-------------+-------+---------------+--------------+---------+-------------+---------------+
| delivery_id | c_id  | customer_name | product_name | company | total_price | address       |
+-------------+-------+---------------+--------------+---------+-------------+---------------+
|      104794 | 10011 | adarsh        | hockeystick  | adidas  |        1500 | gc vihar      |
|      123456 | 10011 | adarsh        | bat          | mrf     |      500000 | ambapua       |
|      123457 | 10012 | ajay          | bat          | mrf     |       56000 | ambapua       |
|      123458 | 10013 | anirudh       | ball         | club    |        2000 | old bus stand |
|      123459 | 10014 | virat         | football     | nike    |        2000 | old bus stand |
|      123460 | 10015 | aayuosh       | football     | adidas  |        3000 | tulsi vihar   |
|      123461 | 10016 | sujal         | badminton    | yonex   |        1500 | tulsi vihar   |
|      123462 | 10017 | sankit        | badminton    | yonex   |        2000 | tata bench    |
|      123463 | 10018 | sidhant       | tennisrecket | adidas  |        2000 | tata bench    |
|      123471 | 10019 | lara          | tennisrecket | adidas  |        5000 | tata bench    |
|      123472 | 10020 | risabh        | football     | adidas  |        5000 | tata bench    |
|      123473 | 10021 | ashutosh      | studs        | nike    |        2000 | tata bench    |
|      123474 | 10022 | yash          | volleyball   | nivea   |        2000 | tulsi vihar   |
|      123475 | 10023 | sukesh        | volleyball   | nivea   |        2000 | tulsi vihar   |
|      123476 | 10024 | zubin         | handball     | nivea   |        1000 | bima nagar    |
+-------------+-------+---------------+--------------+---------+-------------+---------------+
```

Product table

```
mysql> select * from product;
+-----+--------------------+-----------+-------------+-------+---------+
| pid | pname              | company   | sport       | price | size    |
+-----+--------------------+-----------+-------------+-------+---------+
| 110 | ball               | kookabura | cricket     |  5000 | regular |
| 111 | ball               | club      | cricket     |   700 | regular |
| 112 | ball               | stumper   | cricket     |    50 | regular |
| 113 | gloves             | ss        | cricket     |   800 | small   |
| 114 | gloves             | ss        | cricket     |  1500 | medium  |
| 115 | gloves             | ss        | cricket     |  3000 | large   |
| 116 | gloves             | Gm        | cricket     |   600 | small   |
| 117 | gloves             | Gm        | cricket     |  1000 | medium  |
| 118 | gloves             | Gm        | cricket     |  2000 | large   |
| 119 | gloves             | MRF       | cricket     |   200 | small   |
| 120 | gloves             | MRF       | cricket     |   500 | medium  |
| 121 | gloves             | MRF       | cricket     |  1000 | large   |
| 122 | jersey             | nike      | cricket     |   500 | small   |
| 123 | jersey             | nike      | cricket     |   800 | medium  |
| 124 | jersey             | nike      | cricket     |  1000 | large   |
| 125 | jersey             | adidas    | cricket     |  1500 | small   |
| 126 | jersey             | adidas    | cricket     |  2000 | medium  |
| 127 | jersey             | adidas    | cricket     |  5000 | large   |
| 128 | pad                | ss        | cricket     |  1000 | small   |
| 129 | pad                | ss        | cricket     |  2000 | medium  |
| 130 | pad                | ss        | cricket     |  2000 | large   |
| 131 | pad                | MRF       | cricket     |  2000 | medium  |
| 132 | pad                | MRF       | cricket     |  5000 | large   |
| 133 | pad                | MRF       | cricket     |   200 | small   |
| 134 | helmet             | MRF       | cricket     |  1000 | small   |
| 135 | helmet             | MRF       | cricket     |  1500 | medium  |
| 136 | helmet             | MRF       | cricket     |  2000 | medium  |
| 137 | helmet             | MRF       | cricket     |  3000 | large   |
+-----+--------------------+-----------+-------------+-------+---------+
```

Returns table

```
mysql> select * from returns;
+-------------+---------------+--------------+-------------+--------------+-----------------------+
| delivery_id | customer_name | product_name | total_price | address      | reason                |
+-------------+---------------+--------------+-------------+--------------+-----------------------+
|      123456 | adarsh        | bat          |      500000 | ambapua      | size problem          |
|      123457 | ajay          | bat          |       56000 | ambapua      | size problem          |
|      123458 | anirudh       | ball         |        2000 | old bus stand | other company         |
|      123459 | virat         | football     |        2000 | old bus stand | other company         |
|      123460 | aayuosh       | football     |        3000 | tulsi vihar  | size problem          |
|      123461 | sujal         | badminton    |        1500 | tulsi vihar  | didt like the product |
|      123462 | sankit        | badminton    |        2000 | tata bench   | didt like the product |
|      123463 | sidhant       | tennisrecket |        1000 | tata bench   | didt like the product |
|      123464 | aman          | volleyball   |        1000 | new bus stand | didt like the product |
|      123465 | jatin         | studs        |        1500 | new bus stand | size problem          |
|      123466 | spandan       | wicket       |        1000 | bima nagar   | other company         |
|      123467 | rohit         | tennisball   |         500 | bima nagar   | other company         |
|      123468 | dhawan        | tennisball   |         500 | bima nagar   | other company         |
|      123469 | kuldeep       | hockeystick  |         600 | subhas nagar | size problem          |
|      123470 | ramesh        | boxinggloves |         600 | subhas nagar | size problem          |
+-------------+---------------+--------------+-------------+--------------+-----------------------+
15 rows in set (0.00 sec)
```

References

In order to work on this project titled –(ONLINE SPORTS SOFTWARE), the following books and websites are referred by me during the various phases of development of the project.

1) Computer Science (Sumita arora Class XI and XII)

2) www.youtube.com

3) www.python.com

Other than the above-mentioned books, the suggestions and supervision of my teacher and my class experience also helped me to develop this software project.