Table of Contents

1.0 Task A – Develop a Product Management System

1.1 Source Code and Explanation in Comment

1.1.1 Main.java

```
package Assignment1;

import java.util.InputMismatchException; // For try catch

import java.util.Scanner; // To scan input

public class Main {

    public static ProductManagementSystem pms = new ProductManagementSystem(); // To use method in ProductManagementSystem.java

    public static Scanner sc = new Scanner(System.in);  // Declare scanner

    public static void main (String[]args) // Arguments in main method

    {

        int selection=0; // Selection of Manager's input in main menu

        boolean validSelection=false, exitProgram=false;

        Menu menus =  new Menu(); // To use method in Menu.java

        while(!exitProgram) //If manager does not chose to exit program, the program will keep looping

   {

            System.out.println("Welcome to the product management system.\n");

            menus.menu1(); //Show Product Code Table

   do {

    try{ //catch mismatch input error and other possible errors

    menus.menu2(); //Show main menu

    selection = sc.nextInt();

    // Validation for Selection

    if (selection < 1 || selection > 6)

    {

        System.out.println("Invalid selection, please input between 1 and 6.");

        validSelection=false;

        menus.menu2();//Show main menu

        selection = sc.nextInt();
    }
    else
    {
        validSelection=true;
    }
    }
    catch(InputMismatchException e)
    {
        System.out.println("Invalid selection, please input an integer number.");
        validSelection = false;
                 sc.next();
    }
    catch(Exception e)
        {
                 System.out.println("Something is error.");
        }
        }while(!validSelection); //The menu will keep looping if the manager does not
input valid selection
   switch (selection)
   {
   case 1: //Manager chose to add new product
    menus.menu1(); //Show Product Code Table
    pms.addProduct();
    break;
   case 2: // Manager chose to update product
    pms.updateProduct();
    break;
   case 3: // Manager chose to delete product
    pms.deleteProduct();
    break;
```

```
    case 4: //Manager chose to display product based on different criteria
      pms.displayProduct();
      break;
    case 5: //Manager chose to display product code table
      menus.menu1();
      break;
    case 6: //Manager chose to exit the program
      System.out.println("Shutting down the system. Have a nice day.");
      exitProgram=true;
      break;
    default:
      System.out.println("Please enter valid selection.");
    }
      }
      }
}
```

1.1.2 ProductManagementSystem.java

```
package Assignment1;

import java.util.ArrayList; // Data structure chose

import java.util.Scanner; // To scan input

import java.util.InputMismatchException; //For try catch

public class ProductManagementSystem {

      private static Scanner sc = new Scanner(System.in); //Declare scanner

      private ArrayList<Product> products = new ArrayList<>(); //Declare arraylist

      private Menu menus =  new Menu(); //To use method in Menu.java


      public ProductManagementSystem() //For initially store product records
```

```
         {

       products = new ArrayList<>(); //Declare product arraylist

       // Add sample data as it is more realistic and more convenient in testing features

       products.add(new Product("M53201024","Malaysia", "Intel i5", "320 GB", "1 TB", 10));

       products.add(new Product("J53201024","Japan", "Intel i5","320 GB", "1 TB", 5));

       products.add(new Product("A53201024","America", "Intel i5", "320 GB", "1 TB", 20));

       products.add(new Product("M73201024","Malaysia", "Intel i7", "320 GB", "1 TB", 10));

       products.add(new Product("J95002048","Japan", "Intel i9","500 GB", "2 GB", 5));

       products.add(new Product("A510244096","America", "Intel i5", "1 TB", "4 GB", 20));

       products.add(new Product("M55002048","Malaysia", "Intel i5", "500 GB", "2 GB", 10));

       products.add(new Product("J510242048","Japan", "Intel i5","1 TB", "2 GB", 5));

       products.add(new Product("M910244096","Malaysia", "Intel i9", "1 TB", "4 GB", 20));

       products.add(new Product("A910241024","America", "Intel i9", "1 TB", "1 TB", 10));

    }


         public void addProduct() //(1) Add New Products

         {

                  boolean productCodeDuplication = false; //validate if the product code already exist in the system

                  boolean validateCountry = false; //validate if the country of the product code is correct such as 'M'

                  boolean validateProcessor = false; //validate if the processor type is correct such as '5'

                  boolean validateHardDisk = false; //validate if the hard disk is correct such as '320'

                  boolean validateInternalMemory = false; //validate if the internal memory is correct such as '1024'
```

```
            boolean validateQuantity = false; //validate the quantity is correct as it is positive
and numeric

            boolean validateSelection = false; //validate if the selection is in the option list
and numeric

            String productCode = "";

            int selection=0;

            do{

    try{ //catch ArrayIndexOutOfBoundsException and other possible error

    try { //catch input mismatch error for selection

    System.out.println("Enter your selection.");

    menus.menu12(); // to chose either add record or exit

    selection = sc.nextInt();

    while(selection!=1 && selection!=2) //validate selection in range

    {

      System.out.println("Please enter either 1 or 2.");

      System.out.println("Enter your selection.");

      menus.menu12(); // ask to choose add record or exit

      selection = sc.nextInt();

    }

      }catch(InputMismatchException e)

      {

            System.out.println("Invalid selection, please input an integer number.");

                  validateSelection=false;

                  sc.next();

      }
```

```
        switch (selection)

        {

        case 1: //(1) Add Records

        do

        {

           Product product = new Product();

        validateSelection = true;

        sc.nextLine(); //Consume new line

        System.out.println("Enter the product code of the new product according to the product
code table. ");

        System.out.print("Product code: ");

        productCode = sc.nextLine().toUpperCase(); //to handle both upper case and lower case
country character (eg:M and m)

        while(productCode.isEmpty()) // validate it is not empty

        {

            System.out.println("The product code should not be empty.");

            System.out.println("Please try again.");

            System.out.println("Enter the product code of the new product according to the product
code table. ");

            System.out.print("Product code: ");

            productCode = sc.nextLine().toUpperCase();

        }

        while(productCode.length() < 9 || productCode.length() > 10 ) //Validate the length of the
product code

        {

            System.out.println("Invalid length. The product code should only consists of 9 - 10
characters.");
```

```
        System.out.println("Please try again.");
        System.out.println("Enter the product code of the new product according to the product code table. ");
        System.out.print("Product code: ");
        productCode = sc.nextLine().toUpperCase();
    }
    do
    {
    for (Product product_ : products)
    {
        if(product_.getProductCode().equals(productCode)) //validate if the code already exist
        {
            System.out.println("The records of the product code is already in the "
                                + "system, please enter unexisting product code.");
            productCodeDuplication = true;
            System.out.println("Please try again.");
            System.out.println("Enter the product code of the new product according to the product code table. ");
            System.out.print("Product code: ");
            productCode = sc.nextLine().toUpperCase();
        }
        else
        {
                    productCodeDuplication=false;
        }
```

```
}

}while(productCodeDuplication);

String stringProductCode = productCode;

//Retrieve the records details from the product code using charAt

char countryCharacter = stringProductCode.charAt(0); //country character of the product code at index 0

char processorCharacter = stringProductCode.charAt(1); //processor character of the product code at index 0

char hardDiskCharacter = stringProductCode.charAt(2); //hard disk character of the product code at index 0

char internalMemoryCharacter=stringProductCode.charAt(5); //default product code length 9

if (productCode.length()==10) /*if the length of the product code is 10, the

   internal memory character will start at index of 6*/

{

   internalMemoryCharacter = stringProductCode.charAt(6);

}



//Set country according to product code

switch (countryCharacter)

{

   case 'M':

      product.setCountry("Malaysia");

      validateCountry = true;

      break;
```

```
    case 'J':

        product.setCountry("Japan");

        validateCountry = true;

        break;


    case 'A':

        product.setCountry("America");

        validateCountry = true;

        break;

    default:

        System.out.println("Invalid country.");

        validateCountry = false;

}

//Set processor type according to product code

switch(processorCharacter) {

    case '5':

        product.setProcessor("Intel i5");

        validateProcessor = true;

        break;


    case '7':

        product.setProcessor("Intel i7");

        validateProcessor = true;

        break;
```

```
      case '9':

        product.setProcessor("Intel i9");

        validateProcessor = true;

        break;


      default:

        System.out.println("Invalid processor type.");

        validateProcessor = false;

    }

    //hard disk character start from index 2 to index 4

    String hardDiskChar = productCode.substring(2,5);

  //Set hard disk capacity according to product code

    switch(hardDiskCharacter) {

      case '3':


          //Validate is 320 correctly typed

          if(hardDiskChar.equals("320"))

          {

            product.setHardDiskCapacity("320 GB");

            validateHardDisk = true;

            break;

          }

          else
```

```
            {
               System.out.println("The hard disk capacity should be 320. Please enter again.");
               validateHardDisk = false;
               break;
            }



      case '5':
                  //Validate is 500 correctly typed
            if(hardDiskChar.equals("500"))
            {
               product.setHardDiskCapacity("500 GB");
               validateHardDisk = true;
               break;
            }
            else
            {
               System.out.println("The hard disk capacity should be 500. Please enter again. ");
               validateHardDisk = false;
               break;
            }
```

```
        case '1':

            String hardDiskChar2 = productCode.substring(2,6);

            //Validate is 1024 correctly typed

          if(hardDiskChar2.equals("1024"))

          {

             product.setHardDiskCapacity("1 TB");

             validateHardDisk = true;

             break;

          }

          else

          {

             System.out.println("The hard disk capacity should be 1024. Please enter again.
");

             validateHardDisk = false;

             break;

          }




       default:

          //If not equal to either 320, 500, 1024

          System.out.println("Invalid hard disk capacity.");

          validateHardDisk = false;

     }

     //Set internal memory capacity according to product code

     switch(internalMemoryCharacter) {
```

```
    case '1':
            // If the length of the product code is 9, then the internal memory
character start from index 6
        if(productCode.length() == 9)
        {
            String memoryCharacter = productCode.substring(5);
            //Validate is 1024 correctly typed
            if(memoryCharacter.equals("1024")){
                product.setInternalMemoryCapacity("1 TB");
                validateInternalMemory = true;
                break;
            }
            else
            {
                System.out.println("The hardDisk should be 1024. Please enter
again.");
                validateInternalMemory = false;
                break;
            }
      }


       // If the length of the product code is 10, then the internal memory character start
from index 6
      else if(productCode.length() == 10){
        String memoryCharacter = productCode.substring(6);
        //Validate is 1024 correctly typed
```

```
                if(memoryCharacter.equals("1024")){
                    product.setInternalMemoryCapacity("1 TB");
                    validateInternalMemory = true;
                    break;
                }else{
                    System.out.println("The internal memory capacity should be 1024. Please
enter again.");
                    validateInternalMemory = false;
                    break;
                }
              }
        case '2':
                // If the length of the product code is 9, then the internal memory character start
from index 6
                if(productCode.length() == 9)
                {
                        String memoryCharacter = productCode.substring(5);
                        //Validate is 2048 correctly typed
                        if(memoryCharacter.equals("2048")){
                                product.setInternalMemoryCapacity("2 GB");
                                validateInternalMemory = true;
                                break;
                        }
                        else
                        {
```

```
                                        System.out.println("The internal memory capacity should be 2048.
Please enter again.");

                                        validateInternalMemory = false;

                                        break;

                              }

             }



              // If the length of the product code is 10, then the internal memory character start from
index 6

             else if(productCode.length() == 10){

                 String memoryCharacter = productCode.substring(6);

                 //Validate is 2048 correctly typed

                 if(memoryCharacter.equals("2048")){

                                        product.setInternalMemoryCapacity("2 GB");

                    validateInternalMemory = true;

                    break;

                 }else{

                    System.out.println("The internal memory capacity should be 2048. Please enter
again.");

                    validateInternalMemory = false;

                    break;

                 }

              }

              break;

          case '4':
```

```
        // If the length of the product code is 9, then the internal memory character start from index 6
        if(productCode.length() == 9)
        {
            String memoryCharacter = productCode.substring(5);
            //Validate is 4096 correctly typed
            if(memoryCharacter.equals("4096")){
                product.setInternalMemoryCapacity("4 GB");
                validateInternalMemory = true;
                break;
            }
            else
            {
                System.out.println("The internal memory capacity should be 4096. Please enter again.");
                validateInternalMemory = false;
                break;
            }
     }

     // If the length of the product code is 10, then the internal memory character start from index 6
     else if(productCode.length() == 10){
       String memoryCharacter = productCode.substring(6);
       //Validate is 4096 correctly typed
       if(memoryCharacter.equals("4096")){
```

```
                        product.setInternalMemoryCapacity("4 GB");
            validateInternalMemory = true;
            break;
          }else
          {
            System.out.println("The internal memory capacity should be 4096. Please enter again.");
            validateInternalMemory = false;
            break;
          }
        }
        break;
      default :
        System.out.println("Invalid internal memory capacity.");
        validateInternalMemory = false;
    }
    //only run if the product code is totally validate
    if(validateSelection && !productCodeDuplication && validateCountry && validateProcessor &&
            validateHardDisk && validateInternalMemory && validateSelection)
            {
    do
    {
            try //catch input mismatch errors and other possible errors
            {
```

```
        //prompt user to input the quantity for the new record

        System.out.print("Enter product quantity: ");

        int quantity = sc.nextInt();

        while(quantity<0) //validate quantity should be positive

        {

            System.out.println("Invalid quantity number. It should be positive, please enter again.");

            System.out.print("Enter product quantity: ");

            quantity = sc.nextInt();

        }

        validateQuantity = true;

        product.setProductCode(productCode); //set product code according to user input

        product.setQuantity(quantity); //set quantity according to user input

    products.add(product); //add product

    System.out.println("Product record successfully added.");

        }

        catch(InputMismatchException e)

        {

            System.out.println("Invalid selection, please input an integer number.");

            validateQuantity=false;

            sc.next();

        }

        catch(Exception e)

        {
```

```
                  System.out.println("Something is error.");

            }

    }while(!validateQuantity);

            }

    //prompt user for not adding the product as it is invalid

    if(productCodeDuplication || !validateCountry || !validateProcessor ||

            !validateHardDisk || !validateInternalMemory || !validateQuantity
|| !validateSelection)

    {

      System.out.println("Product failed to add, please check again the product code.");

    }

    break;

    }while(productCodeDuplication || !validateCountry || !validateProcessor ||

            !validateHardDisk || !validateInternalMemory || !validateQuantity
|| !validateSelection);



    case 2: //(2) Exit

      validateSelection = true;

      break;

    }

    }

    catch(ArrayIndexOutOfBoundsException e){

      System.out.println("Error, product code has not added, please try again.");

    }

    catch(Exception e)
```

```
                {

                        System.out.println("Something is error.");

                }

  }while(!validateSelection || selection!=2);

                //will keep looping if either selection is invalid or user not chose to exit

        }


        public void updateProduct() //(2) Update Records

        {

                boolean validSelection = false;

                int selection=0;

                do

                {

                try // catch input mismatch error and other possible errors

                {

                boolean validSelection2=false;

           menus.menu8();

           selection = sc.nextInt();

           switch(selection)

           {

           case 1: // Update (1) Manufacturing Country

                do

                {

                try //catch mismatch input error and other possible errors
```

```
            {
            int selecCount1=0, selecCount2=0, updateSelection = 0; /*selecCount1 is old
country option
            selecCount2 is new country option*/
            int count=0; //to calculate how many records equal to the old country
            String originalCountry="", newCountry="";
            do
            {
            do
            {
            System.out.println("Select the manufacturing country of the product you want to
update.");
            menus.menu4(); //display country for user to select
            selecCount1 = sc.nextInt();
            if(selecCount1<1||selecCount1>3)
            {
                  System.out.println("Invalid selection. Please enter between 1 - 3.");
            }
            }while(selecCount1<1||selecCount1>3);
            do
            {
            System.out.println("Select the new manufacturing country:");
            menus.menu4();
            selecCount2 = sc.nextInt();
            if(selecCount2<1||selecCount2>3)
            {

                  System.out.println("Invalid selection. Please enter between 1 - 3.");

            }

            }while(selecCount2<1||selecCount2>3);

            if (selecCount1==selecCount2) //validate if old and new is same

            {

                  System.out.println("The old and new manufacturing country should not be the same. Please try again.");

            }

            }while(selecCount1==selecCount2);

            switch(selecCount1) //assign original country according to option

            {

            case 1:

                  originalCountry = "Malaysia";

                  break;



            case 2:

                  originalCountry = "Japan";

                   break;



            case 3:

                  originalCountry = "America";

                  break;



            default:
```

```
            System.out.println("Invalid selection. Please try again.");
            validSelection2 = false;
        }
        switch(selecCount2) //assign new country according to option
        {
        case 1:
            newCountry = "Malaysia";
            break;

        case 2:
            newCountry = "Japan";
             break;

        case 3:
            newCountry = "America";
            break;

        default:
            System.out.println("Invalid selection. Please try again.");
            validSelection2 = false;
        }
        for (Product product : products) //count number of records same as original country
    {
        if(product.getCountry().equals(originalCountry))
```

```
                {

                    count++;

                }

            }

                        System.out.println("There are "+count+" records with manufacturing country of "+originalCountry+":");

                        if (count!=0) // if at least one record will display the record that is old country

                        {

                        menus.menu(); //print header

                        for (Product product : products) //display records that are same as original country

            {

                        if(product.getCountry().equals(originalCountry))

                {

                    System.out.print(product);

                }

            }

                        menus.menu9(); //ask either update all or update one

                        updateSelection = sc.nextInt();

                        switch(updateSelection)

                        {

                        case 1: // (1) Update All

                                    for (Product product : products) //set every product that are original country to new country

                            {

                                if (product.getCountry().equals(originalCountry))
```

```
     {
        product.setCountry(newCountry);
     }
   }
        System.out.println("The product records has been update.");
        break;

case 2: // (2) Update One
        int cont=0; //for user to continue update or not
        do
        {
        Product productX = null; // to store the product
        String productCode="";
        boolean validProductCode = true, validProduct = true;
        do
        {
                System.out.println("Please enter the product code you want to update.");
                System.out.print("Product code: ");
                if(validProductCode && validProduct)
                {
                        sc.nextLine(); //Consume new line
                }
                productCode = sc.nextLine().toUpperCase();
```

```
            for (Product product : products) //validate if the product is in the
system

          {

           if (!product.getProductCode().equals(productCode))

           {

            validProductCode = false;

           }

           else

           {

            productX = product;

            validProductCode = true;

              break;

           }

          }

            //if the product code is validate but the product country is already
the new country

            if (validProductCode && productX != null &&
productX.getCountry().equals(newCountry))

            {

               System.out.println("The country of "+productCode+" is
already "+newCountry+".");

               validProduct = false;

            }

            else

            {

               validProduct = true;
```

```
                }

                if(!validProductCode)

                {

                    System.out.println("Product code not exist, please add the product to the system first.");

                }

            }while(!validProductCode || !validProduct); //will keep looping if the product or product code invalid

            for (Product product : products) //update the product country to new country

         {

                if (product.getProductCode().equals(productCode))

           {

            product.setCountry(newCountry);

           }

         }

            System.out.println(productCode+" successfully updated.");

            System.out.println("Continue update?");

            menus.menu10(); //ask user to select yes or no

            cont = sc.nextInt();

            }while(cont!=2); //will keep looping until user choose no for the continue option

            break;


        default:

            System.out.println("Invalid selection. Please try again.");
            validSelection2 = false;
        }
        }
        validSelection2=true;
        }catch(InputMismatchException e)
            {
                System.out.println("Invalid selection, please input an integer
number.");
                validSelection=false;
                sc.next();
            }
        catch(Exception e)
            {
                System.out.println("Something is error.");
            }
            }while(!validSelection2);
        break;

    case 2: // Update (2) Processor Type
        do
        {
        try //catch mismatch input error and other possible errors
        {
        int selecCount1=0, selecCount2=0, updateSelection = 0; //selecCount1 is old
option selecCount2 is new option
```

```
int count=0; //count product with original processor
String originalProcessor="", newProcessor="";
do
{
do
{
System.out.println("Select the processor type of the product you want to update.");
menus.menu5(); //ask user to choose original processor type i5 i7 i9
selecCount1 = sc.nextInt();
if(selecCount1<1||selecCount1>3)
{
        System.out.println("Invalid selection. Please enter between 1 - 3.");
}
}while(selecCount1<1||selecCount1>3);
do
{
System.out.println("Select the new processor type:");
menus.menu5(); //ask user to choose new processor type i5 i7 i9
selecCount2 = sc.nextInt();
if(selecCount2<1||selecCount2>3)
{
        System.out.println("Invalid selection. Please enter between 1 - 3.");
}
}while(selecCount2<1||selecCount2>3);
```

```
        if (selecCount1==selecCount2) //validate if original and new is same option
        {
                System.out.println("The old and new processor type should not be the same. Please try again.");
        }
        }while(selecCount1==selecCount2);
        switch(selecCount1) //assign original according to option
        {
        case 1:
                originalProcessor = "Intel i5";
                break;

        case 2:
                originalProcessor = "Intel i7";
                 break;

        case 3:
                originalProcessor = "Intel i9";
                break;

        default:
                System.out.println("Invalid selection. Please try again.");
                validSelection2 = false;
        }
        switch(selecCount2) //assign new according to option
        {
        case 1:
                newProcessor = "Intel i5";
                break;

        case 2:
                newProcessor = "Intel i7";
                 break;

        case 3:
                newProcessor = "Intel i9";
                break;

        default:
                System.out.println("Invalid selection. Please try again.");
                validSelection2 = false;
        }
        for (Product product : products) //count product same as original processor
{
        if(product.getProcessor().equals(originalProcessor))
   {
      count++;
   }
}
```

```
                System.out.println("There are "+count+" records with processor type of
"+originalProcessor+":");

                if (count!=0)

                {

                menus.menu(); //print header

                for (Product product : products) //display record with same original processor

        {

                        if(product.getProcessor().equals(originalProcessor))

           {

              System.out.print(product);

           }

        }

                menus.menu9(); //ask update all or one

                updateSelection = sc.nextInt();

                switch(updateSelection)

                {

                case 1: // (1) Update All

                        for (Product product : products)

                   {

                                if(product.getProcessor().equals(originalProcessor))

                      {

                        product.setProcessor(newProcessor);

                      }

                   }

                        System.out.println("The product records has been update.");
```

```
            break;

        case 2:

            int cont=0; // for user choose to continue or not

            do

            {

                Product productX = null; //store product

                String productCode="";

                boolean validProductCode = true, validProduct = true;

            do

            {

                System.out.println("Please enter the product code you want to
update.");

                System.out.print("Product code: ");

                if(validProductCode && validProduct)

                {

                    sc.nextLine(); //Consume new line

                }

                productCode = sc.nextLine().toUpperCase();

                for (Product product : products) //check if product code is validate

             {

               if (!product.getProductCode().equals(productCode))

               {

                 validProductCode = false;

               }

               else
```

```
           {

            productX = product;

            validProductCode = true;

              break;

           }

          }

            //if product code is validate but old same as new

            if (validProductCode && productX != null && 
productX.getProcessor().equals(newProcessor))

            {

               System.out.println("The processor type of 
"+productCode+" is already "+newProcessor+".");

               validProduct = false;

            }

            else

            {

               validProduct = true;

            }

            if(!validProductCode)

            {

               System.out.println("Product code not exist, please add the 
product to the system first.");

            }

         }while(!validProductCode || !validProduct);

         for (Product product : products) //set product to new processor

       {
```

```
                        if (product.getProductCode().equals(productCode))
                    {
                      product.setProcessor(newProcessor);
                    }
                  }
                      System.out.println(productCode+" successfully updated.");
                      System.out.println("Continue update?");
                      menus.menu10();
                      cont = sc.nextInt();
                      }while(cont!=2); //will keep looping until user choose no for continue
option
                      break;


                default:
                      System.out.println("Invalid selection. Please try again.");
                      validSelection2 = false;
                }
                }
                validSelection2 = true;
                }catch(InputMismatchException e)
                      {
                            System.out.println("Invalid selection, please input an integer
number.");
                            validSelection=false;
                            sc.next();
```

```
            }
        catch(Exception e)
            {
                System.out.println("Something is error.");
            }
            }while(!validSelection2);
        break;


    case 3: // Update (3) Hard Disk Capacity
        do
        {
        try //catch mismatch input error and other possible errors
        {
        int selecCount1=0, selecCount2=0, updateSelection = 0; //selecCount1 is old
hard disk selecCount2 is new
        int count=0; //count product with same original hard disk
        String originalHardDisk="", newHardDisk="";
        do
        {
        do
        {
        System.out.println("Select the hard disk capacity of the product you want to
update.");
        menus.menu6(); //ask to select old capacity 320 500 1024
        selecCount1 = sc.nextInt();
        if(selecCount1<1||selecCount1>3)

        {

            System.out.println("Invalid selection. Please enter between 1 - 3.");

        }

        }while(selecCount1<1||selecCount1>3);

        do

        {

        System.out.println("Select the new hard disk capacity:");

        menus.menu6(); //ask to select new capacity 320 500 1024

        selecCount2 = sc.nextInt();

        if(selecCount2<1||selecCount2>3)

        {

            System.out.println("Invalid selection. Please enter between 1 - 3.");

        }

        }while(selecCount2<1||selecCount2>3);

        if (selecCount1==selecCount2) //check if old and new is same

        {

            System.out.println("The old and new hard disk capacity should not be the
same. Please try again.");

        }

        }while(selecCount1==selecCount2);

        switch(selecCount1) //assign old hard disk according to option

        {

        case 1:

            originalHardDisk = "320 GB";
```

```
    break;


case 2:

    originalHardDisk = "500 GB";

     break;


case 3:

    originalHardDisk = "1 TB";

    break;


default:

    System.out.println("Invalid selection. Please try again.");

    validSelection2 = false;

}

switch(selecCount2) //assign new hard disk according to option

{

case 1:

    newHardDisk = "320 GB";

    break;


case 2:

    newHardDisk = "500 GB";

     break;
```

```
                case 3:

                        newHardDisk = "1 TB";

                        break;



                default:

                        System.out.println("Invalid selection. Please try again.");

                        validSelection2 = false;



                }

                for (Product product : products) //count product same as original hard disk

        {

                if(product.getHardDiskCapacity().equals(originalHardDisk))

           {

              count++;

           }

        }

                System.out.println("There are "+count+" records with hard disk capacity of
"+originalHardDisk+":");

                if (count!=0)

                {

                menus.menu(); //print header

                for (Product product : products) //display product with same hard disk as original

        {

                        if(product.getHardDiskCapacity().equals(originalHardDisk))

           {
                System.out.print(product);

            }

        }

                menus.menu9(); //ask to update all or one

                updateSelection = sc.nextInt();

                switch(updateSelection)

                {

                case 1: //(1) Update All

                        for (Product product : products)

                    {

                                if(product.getHardDiskCapacity().equals(originalHardDisk))

                        {

                            product.setHardDiskCapacity(newHardDisk);

                        }

                    }

                        System.out.println("The product records has been update.");

                        break;

                case 2:

                        int cont=0; //for user to choose whether continue update or not

                        do

                        {

                                Product productX = null; //to store product

                                String productCode="";

                                boolean validProductCode = true, validProduct = true;
                        do

                        {

                                System.out.println("Please enter the product code you want to
                update.");

                                System.out.print("Product code: ");

                                if(validProductCode && validProduct)

                                {

                                    sc.nextLine(); //Consume new line

                                }

                                productCode = sc.nextLine().toUpperCase();

                                for (Product product : products)  //validate productCode exist or
                not

                             {

                               if (!product.getProductCode().equals(productCode))

                               {

                                validProductCode = false;

                               }

                               else

                               {

                                productX = product;

                                validProductCode = true;

                                   break;

                               }

                             }

                                // if productCode valid but original same with new hard disk
                                if (validProductCode && productX != null &&
                productX.getHardDiskCapacity().equals(newHardDisk))

                                {

                                    System.out.println("The hard disk capacity of
                "+productCode+" is already "+newHardDisk+".");

                                    validProduct = false;

                                }

                                else

                                {

                                    validProduct = true;

                                }

                                if(!validProductCode)

                                {

                                    System.out.println("Product code not exist, please add the
                product to the system first.");

                                }

                        }while(!validProductCode || !validProduct);

                        for (Product product : products)  //set product hard disk to new hard disk

                     {

                                if (product.getProductCode().equals(productCode))

                           {

                        product.setHardDiskCapacity(newHardDisk);

                           }

                     }

                        System.out.println(productCode+" successfully updated.");

                        System.out.println("Continue update?");
```

```
                menus.menu10(); //ask user yes or no
                cont = sc.nextInt();
                }while(cont!=2); //will keep looping until user choose exit option
                break;


        default:
                System.out.println("Invalid selection. Please try again.");
                validSelection2 = false;
        }
        }
        validSelection2 = true;
        }catch(InputMismatchException e)
                {
                        System.out.println("Invalid selection, please input an integer number.");
                        validSelection=false;
                        sc.next();
                }
        catch(Exception e)
                {
                        System.out.println("Something is error.");
                }
                }while(!validSelection2);
        break;
```

```
    case 4: // Update (4) Internal Memory Capacity

        do

        {

        try

        {

        int selecCount1=0, selecCount2=0, updateSelection = 0; //selecCount1 is old
capacity option selecCount2 is new

        int count=0; //for counting numbers of product same as original capacity

        String originalInternalMemory="", newInternalMemory="";

        do

        {

        do

        {

        System.out.println("Select the internal memory capacity of the product you want
to update.");

        menus.menu7(); //ask user to select old capacity

        selecCount1 = sc.nextInt();

        if(selecCount1<1||selecCount1>3)

        {

            System.out.println("Invalid selection. Please enter between 1 - 3.");

        }

        }while(selecCount1<1||selecCount1>3);

        do

        {

        System.out.println("Select the new internal memory capacity:");
        menus.menu7(); //ask user to select new capacity

        selecCount2 = sc.nextInt();

        if(selecCount2<1||selecCount2>3)

        {

            System.out.println("Invalid selection. Please enter between 1 - 3.");

        }

        }while(selecCount2<1||selecCount2>3);

        if (selecCount1==selecCount2) //validate if old and new option are the same

        {

            System.out.println("The old and new internal memory capacity should not be the same. Please try again.");

        }

        }while(selecCount1==selecCount2);

        switch(selecCount1) //assign old capacity according to option

        {

        case 1:

            originalInternalMemory = "1 TB";

            break;



        case 2:

            originalInternalMemory = "2 GB";

             break;



        case 3:

            originalInternalMemory = "4 GB";
            break;


        default:
            System.out.println("Invalid selection. Please try again.");
            validSelection2 = false;
        }
        switch(selecCount2) //assign new capacity according to option
        {
        case 1:
            newInternalMemory = "1 TB";
            break;


        case 2:
            newInternalMemory = "2 GB";
             break;


        case 3:
            newInternalMemory = "4 GB";
            break;


        default:
            System.out.println("Invalid selection. Please try again.");
            validSelection2 = false;
        }
```

```
                for (Product product : products) //count product same as original internal memory capacity

        {

                if(product.getInternalMemoryCapacity().equals(originalInternalMemory))

           {

              count++;

           }

        }

                System.out.println("There are "+count+" records with internal memory capacity of "+originalInternalMemory+":");

                if (count!=0)

                {

                menus.menu(); //print headers

                for (Product product : products) //display product same as original internal memory

        {

                        if(product.getInternalMemoryCapacity().equals(originalInternalMemory))

           {

              System.out.print(product);

           }

        }

                menus.menu9(); //ask to update all or one

                updateSelection = sc.nextInt();

                switch(updateSelection)

                {

                case 1: //(1) Update All
```

```
                        for (Product product : products)
                    {
        if(product.getInternalMemoryCapacity().equals(originalInternalMemory))
                      {
                        product.setInternalMemoryCapacity(newInternalMemory);
                      }
                    }
                        System.out.println("The product records has been update.");
                        break;

                case 2:
                        int cont=0; //for user to choose whether continue or not
                        do
                        {
                                Product productX = null; //to store product
                                String productCode="";
                                boolean validProductCode = true, validProduct = true;
                        do
                        {
                                System.out.println("Please enter the product code you want to
update.");
                                System.out.print("Product code: ");
                                if(validProductCode && validProduct)
                                {
```

```
                                             sc.nextLine(); //Consume new line

                                    }

                                    productCode = sc.nextLine().toUpperCase();

                                    for (Product product : products)  //to validate if product code exist

                              {

                                 if (!product.getProductCode().equals(productCode))

                                 {

                                    validProductCode = false;

                                 }

                                 else

                                 {

                                    productX = product;

                                    validProductCode = true;

                                          break;

                                 }

                              }

                                    //if product code exist but original memory same as new memory

                                    if (validProductCode && productX != null &&
productX.getInternalMemoryCapacity().

                                                      equals(newInternalMemory))

                                    {

                                             System.out.println("The internal memory capacity of
"+productCode+" is already "+newInternalMemory+".");

                                             validProduct = false;

                                    }
```

```
                                else
                                {
                                        validProduct = true;
                                }
                                if(!validProductCode)
                                {
                                        System.out.println("Product code not exist, please add the product to the system first.");
                                }
                        }while(!validProductCode || !validProduct);
                        for (Product product : products)  //set product memory to new memory
                  {
                                if (product.getProductCode().equals(productCode))
                     {
                        product.setInternalMemoryCapacity(newInternalMemory);
                     }
                  }
                        System.out.println(productCode+" successfully updated.");
                        System.out.println("Continue update?");
                        menus.menu10(); //ask user to choose yes or no
                        cont = sc.nextInt();
                        }while(cont!=2); //will keep looping until user choose no for continue option
                        break;
                default:
```

```
            System.out.println("Invalid selection. Please try again.");

            validSelection2 = false;

        }

        }

        validSelection2=true;

        }catch(InputMismatchException e)

            {

                System.out.println("Invalid selection, please input an integer number.");

                validSelection=false;

                sc.next();

            }

        catch(Exception e)

            {

                System.out.println("Something is error.");

            }

            }while(!validSelection2);

        break;


      case 5: // Update (5) Quantity

        do

        {

        try

        {

            int quant1, quant2, updateSelection, count=0;
            /*quant1=old quantity

             quant2=new

            update selection = update all or one

            count use to count product with same old quantity*/

            boolean validQuantity = false;

            do

            {

            do

            {

            System.out.println("Enter the quantity of the product you want to update.");

            System.out.print("Quantity: ");

            quant1 = sc.nextInt();

            if(quant1<0)

            {

                System.out.println("Invalid quantity, the quantity should not be negative.");

            }

            }while(quant1<0);

            for (Product product : products) //check whether there are record with old quantity

        {

                if(product.getQuantity()!=quant1)

         {

           validQuantity = false;

         }
```

```
                else
                {
                    validQuantity = true;
                    break;
                }
        }
            if (!validQuantity)
            {
                System.out.println("No records with quantity of "+quant1+". Please
enter again.");
            }
            }while(!validQuantity);
            do
            {
            System.out.println("Enter the new quantity:");
            System.out.print("Quantity: ");
            quant2 = sc.nextInt();
            if(quant2<0)
            {
                System.out.println("Invalid quantity, the quantity should not be
negative.");
            }
            }while(quant2<0);
            if (quant1==quant2) //check if old and new quantity are the same
            {
```

```
                                        System.out.println("The old and new quantity should not be the 
same. Please try again.");
                              }
                              for (Product product : products) //count for the product with old quantity
                    {
                              if(product.getQuantity()==quant1)
                         {
                           count++;
                         }
                    }
                              System.out.println("There are "+count+" records with quantity of 
"+quant1+":");
                              if (count!=0)
                              {
                              menus.menu(); //print header
                              for (Product product : products) //display product with old quantity
                    {
                                        if(product.getQuantity()==quant1)
                         {
                           System.out.print(product);
                         }
                    }
                              }
                              menus.menu9(); //ask to update all or one
                              updateSelection = sc.nextInt();
```

```
                        switch(updateSelection)
                        {
                        case 1: //(1) Update All
                                for (Product product : products)
                           {
                                        if(product.getQuantity()==quant1)
                             {
                                product.setQuantity(quant2);
                             }
                           }
                                System.out.println("The product records has been update.");
                                break;

                        case 2: //(2) Update one
                                int cont=0; //use for user to choose continue update or not
                                do
                                {
                                        Product productX = null; //to store product
                                        String productCode="";
                                        boolean validProductCode = true, validProduct = true;
                                do
                                {
                                        System.out.println("Please enter the product code you
want to update.");
                                        System.out.print("Product code: ");
                                        if(validProductCode && validProduct)
                                        {
                                                sc.nextLine(); //Consume new line
                                        }
                                        productCode = sc.nextLine().toUpperCase();
                                        for (Product product : products) //validate product code
exist or not
                    {
                      if (!product.getProductCode().equals(productCode))
                      {
                                        validProductCode = false;
                      }
                      else
                      {
                                        productX = product;
                                        validProductCode = true;
                                            break;
                      }
                    }
                                        //if product code validate but its quantity same as new
quantity
                                        if (validProductCode && productX != null &&
productX.getQuantity()==(quant2))
                                        {
                                                System.out.println("The quantity of
"+productCode+" is already "+quant2+".");
```

```
                    validProduct = false;
                }
                else
                {
                    validProduct = true;
                }
                if(!validProductCode)
                {
                    System.out.println("Product code not exist, please add the product to the system first.");
                }
            }while(!validProductCode || !validProduct);
            for (Product product : products) //set product quantity to new quantity
        {
                if (product.getProductCode().equals(productCode))
          {
            product.setQuantity(quant2);
          }
        }
            System.out.println(productCode+" successfully updated.");
            System.out.println("Continue update?");
            menus.menu10(); //ask for user to choose yes or no
            cont = sc.nextInt();
            }while(cont!=2); //will keep looping until user choose no for continue option
            break;
         default:

            System.out.println("Invalid selection. Please try again.");
            validSelection2 = false;
         }
      validSelection2=true;
      }catch(InputMismatchException e)
         {
            System.out.println("Invalid selection, please input an integer number.");
            validSelection=false;
            sc.next();
         }
      catch(Exception e)
         {
            System.out.println("Something is error.");
         }
         }while(!validSelection2);
      break;

    case 6: // (6) Exit Update
      validSelection = true;
      break;
```

```
  default:
System.out.println("Please enter valid selection.");
  }
    }catch(InputMismatchException e)
    {
        System.out.println("Invalid selection, please input an integer number.");
        validSelection=false;
        sc.next();
    }
    catch(Exception e)
    {
        System.out.println("Something is error.");
    }
    }while(selection!=6 || !validSelection);
}

public void deleteProduct()
{
    int selection = 0;
    do
    {
        String productCode;
        boolean validSelection = false;
        try //catch mismatch input error and other possible errors
```

```
{
System.out.println("Enter your selection.");
menus.menu11(); //ask user to choose delete record or leave
selection = sc.nextInt();
while(selection!=1 && selection!=2) //check if selection in range
{
	System.out.println("Invalid selection, please input either 1 or 2.");
	System.out.println("Enter your selection.");
	menus.menu11();
	selection = sc.nextInt();
}
switch(selection)
{
case 1: //(1) Delete Records
	boolean validProductCode = true;
	int x = 0, z=1; //to check whether need to consume new line
	do
	{
		try //catch mismatch input error and other possible errors
		{
	int sel=0, del=0; //sel is selection whether want to delete, del is the
index of the product need to delete
	System.out.println("Enter the product code of the record that you
want to remove.");
	System.out.print("Product Code: ");
```

```
                if(validProductCode|| ((x!=0)&&(validProductCode))||
(!validProductCode && z==0))
                {
                    sc.nextLine(); //Consume new line
                }
                productCode = sc.nextLine().toUpperCase();
                for (Product product_ : products)
          {
            if(product_.getProductCode().equals(productCode)) //find the record that
need to be delete
            {
              System.out.println(productCode+" is found.");
              validProductCode = true;
              System.out.println("Are you sure you want to delete it permanently?");
              menus.menu10(); //ask user to choose yes or no
              sel = sc.nextInt();
              if(sel==1) //(1) Yes, which is delete record
              {
                validSelection = true;
               del = products.indexOf(product_); //get the index of product that
need to delete
               z++;
              }
              else if (sel==2)
              {
                validSelection = true;
```

```
                    x=0;

                    System.out.println(productCode+" has not deleted.");

                    x++; /*variable x is used to determine whether need to print the
line for consume new line

                    the reason is it can check whether the code goes through this
line*/

                    z++;

                    break;

                 }

                 else {

                    {

                         z=0;

                         validSelection = false;

                         System.out.println("Invalid selection. Please input either 1
or 2.");

                    }

                              }

               }

               else

               {

                    validSelection = false;

                    validProductCode = false; //to make the consume new line
condition work

               }

             }

                    if(sel==1) //if user choose to delete
```

```
                {
                    products.remove(del); //delete the product
                    System.out.println(productCode+" successfully deleted.");
                    break;
                }
                if(!validProductCode)
        {
            System.out.println("The product code does not exist in the system.");
        }
                    }catch(InputMismatchException e)
                    {
                        System.out.println("Invalid selection, please input
an integer number.");
                        validSelection=false;
                        sc.next();
                    }
                    catch(Exception e)
                    {
                        System.out.println("Something is error.");
                    }
                }while(!validProductCode || !validSelection); /*will keep looping if
either invalid productCode
                or invalid selection*/
                break;
            case 2: // (2) No, which is cancel deletion
                break;
            }
            }catch(InputMismatchException e)
            {
                System.out.println("Invalid selection, please input an integer number.");
                validSelection=false;
                sc.next();
            }
            catch(Exception e)
            {
                System.out.println("Something is error.");
            }
        }while(selection!=2);
    }

    public void displayProduct()
    {
        int displaySelection=0, countrySelection, processorSelection, hardDiskSelection,
        internalMemorySelection, x,y,z;
        boolean validSelection = false, validSelection2 = false;
        do
        {
        try { //catch mismatch input error and other possible errors
```

```
menus.menu3(); //display display menu

displaySelection = sc.nextInt();

while(displaySelection < 1 || displaySelection > 7) //validate selection in range

{

  System.out.println("Invalid selection. Please input between 1 and 7.");

  menus.menu3(); //display display menu

  displaySelection = sc.nextInt();

}

switch(displaySelection)

{

  case 1 :  //(1) Display According to Manufacturing Country

        do

        {

        try //catch input mismatch error and other possible errors

        {

        validSelection=true;

        System.out.println("Select the Manufacturing Country.");

        menus.menu4(); //display country menu

    countrySelection = sc.nextInt();

    while(countrySelection < 1 || countrySelection > 3) //validate selection in range

    {

      System.out.println("Invalid selection. Please input between 1 and 3.");

      System.out.println("Select the Manufacturing Country.");

      menus.menu4(); //display country menu
```

```
            countrySelection = sc.nextInt();

        }

        switch(countrySelection)

        {

            case 1: //(1) Malaysia

                validSelection2=true;

                x=0; //to check if got product equals to Malaysia

                System.out.println("The product records with manufacturing country of Malaysia");

                for(Product product : products){

                    if(product.getCountry().equals("Malaysia"))

                    {

                        x++;

                    }

                }

                if(x==0)

                {

                        System.out.println("No record with manufacturing country of Malaysia.");

                }

                else if (x!=0)

                {

                        menus.menu(); //print headers

                        for (Product product : products) //print product which country equals to Malaysia

                        {
```

```
                        if(product.getCountry().equals("Malaysia"))
                   {
                      System.out.print(product);
                   }
                     }
              }
              break;


            case 2: //(2) Japan
              validSelection2=true;
              y=0; //to check if got product equals to Japan
              System.out.println("The product records with manufacturing country of Japan");
              for(Product product : products)
              {
                 if(product.getCountry().equals("Japan")){
                    y++;
                 }
              }
              if (y==0)
              {
                     System.out.println("No record with manufacturing country of Japan.");
              }
              else if (y!=0)
              {
```

```
                                 menus.menu(); //print headers

                                 for (Product product : products) //print products with country equals to
Japan

                                 {

                                            if(product.getCountry().equals("Japan"))

                              {

                                  System.out.print(product);

                              }

                                 }

                      }

                      break;


                   case 3: //(3) America

                      validSelection2=true;

                      z=0; //to check if got product equals to America

                      System.out.println("The product records with manufacturing country of America");

                      for(Product product : products)

                      {

                          if(product.getCountry().equals("America")){

                              z++;

                          }

                      }

                      if(z==0)

                      {

                                 System.out.println("No record with manufacturing country of America.");
```

```
        }

        else if (z!=0)

        {

            menus.menu(); //print headers

            for (Product product : products) //print records that country equals to America

            {

                if(product.getCountry().equals("America"))

           {

             System.out.print(product);

           }

            }

        }

        break;


       default:

        System.out.println("Invalid selection. Please try again.");

        System.exit(countrySelection);

     }

        }catch(InputMismatchException e)

        {

            System.out.println("Invalid selection, please input an integer number.");

            validSelection2=false;

            sc.next();

        }
```

```
            catch(Exception e)

            {

                        System.out.println("Something is error.");

            }

            }while(!validSelection2);

      break;


   case 2: //(2) Display According to Processor Type

            do

            {

            try //catch input mismatch error and other possible errors

            {

            validSelection=true;

            x=0; //to check if got product equals to i5

            y=0; //to check if got product equals to i7

            z=0; //to check if got product equals to i9

            System.out.println("Select the Processor Type.");

      menus.menu5();//display processor type menu

      processorSelection = sc.nextInt();

      while(processorSelection < 1 || processorSelection > 3) //validate selection in range

      {

         System.out.println("Invalid selection. Please input between 1 and 3.");

                        System.out.println("Select the Processor Type.");

         menus.menu5(); //display processor type menu
```

```
            processorSelection = sc.nextInt();

        }

        switch(processorSelection)

        {

            case 1: //(1) Intel i5

                validSelection2=true;

                System.out.println("The product records with processor type of Intel i5");

                for(Product product : products)

                {

                    if(product.getProcessor().equals("Intel i5"))

                    {

                        x++;

                    }

                }

                if (x==0)

                {

                            System.out.println("No record with processor type of Intel i5.");

                }

                else if (x!=0)

                {

                            menus.menu(); //print headers

                            for (Product product : products)

                            {

                                        if(product.getProcessor().equals("Intel i5")) //print records that
processor i5
```

```
                {

                   System.out.print(product);

                }

                   }

            }

            break;


         case 2: //(2) Intel i7

           validSelection2=true;

           System.out.println("The product records with processor type of Intel i7");

            for(Product product : products)

            {

                if(product.getProcessor().equals("Intel i7"))

                {

                    y++;

                }

            }

            if (y==0)

            {

                      System.out.println("No record with processor type of Intel i7.");

            }

            else if (y!=0)

            {

                      menus.menu(); //print headers
```

```
               for (Product product : products) //print records that processor i7

               {

                           if(product.getProcessor().equals("Intel i7"))

            {

                System.out.print(product);

            }

               }

    }

    break;


case 3: //(3) Intel i9

   validSelection2=true;

    System.out.println("The product records with processor type of Intel i9");

    for(Product product : products)

    {

        if(product.getProcessor().equals("Intel i9"))

        {

            z++;

        }

    }

    if (z==0)

    {

               System.out.println("No record with processor type of Intel i9.");

    }
```

```
        else if (z!=0)
        {
                    menus.menu(); //print headers
                    for (Product product : products) //print records with processor i9
                    {
                                if(product.getProcessor().equals("Intel i9"))
                {
                     System.out.print(product);
                }
                    }
        }
        break;


    default:
        System.out.println("Invalid selection. Please try again.");
        System.exit(processorSelection);
}
        }catch(InputMismatchException e)
        {
                    System.out.println("Invalid selection, please input an integer number.");
                    validSelection2=false;
                    sc.next();
        }
        catch(Exception e)
```

```
                {

                        System.out.println("Something is error.");

                }

                }while(!validSelection2);

           break;


        case 3 : //(3) Display According to Hard Disk Capacity

                do

                {

                try //catch input mismatch error and other possible errors

                {

                validSelection=true;

                x=0; //to check if got product equals to 320 GB

                y=0; //to check if got product equals to 500 GB

                z=0; //to check if got product equals to 1 TB

                System.out.println("Select the Hard Disk Capacity.");

           menus.menu6(); //display hard disk menu

           hardDiskSelection = sc.nextInt();

           while(hardDiskSelection < 1 || hardDiskSelection > 3) //validate selection range

           {

              System.out.println("Invalid selection. Please input between 1 and 3.");

                        System.out.println("Select the Hard Disk Capacity.");

              menus.menu6(); //display hard disk menu

              hardDiskSelection = sc.nextInt();
```

```
}

switch(hardDiskSelection)

{

    case 1: //(1) 320 GB

       validSelection2=true;

        System.out.println("The product records with with hard disk capacity of 320 GB");

        for(Product product : products)

        {

            if(product.getHardDiskCapacity().equals("320 GB"))

            {

                x++;

            }

        }

        if (x==0)

        {

                    System.out.println("No record with hard disk capacity of 320 GB.");

        }

        else if (x!=0)

        {

                    menus.menu(); //print headers

                    for (Product product : products) //print products with capacity of 320 GB

                    {

                                if(product.getHardDiskCapacity().equals("320 GB"))

                {
                     System.out.print(product);
                  }
                    }
              }
              break;


           case 2 : //(2) 500 GB
             validSelection2=true;
              System.out.println("The records with hard disk equals to 500 GB");
              for(Product product : products)
              {
                 if(product.getHardDiskCapacity().equals("500 GB"))
                 {
                    y++;
                 }
              }
              if (y==0)
              {
                     System.out.println("No record with hard disk capacity of 500 GB.");
              }
              else if (y!=0)
              {
                     menus.menu(); //print headers
                     for (Product product : products) //print products with capacity of 500 GB
```

```
               {
                           if(product.getHardDiskCapacity().equals("500 GB"))
            {
                System.out.print(product);
            }
               }
    }
    break;

case 3 : //(3) 1 TB
   validSelection2=true;
    System.out.println("The records with hard disk equals to 1 TB");
    for(Product product : products)
    {
        if(product.getHardDiskCapacity().equals("1 TB"))
        {
            z++;
        }
    }
    if (z==0)
    {
               System.out.println("No record with hard disk capacity of 1 TB.");
    }
    else if (z!=0)
```

```
                {
                        menus.menu(); //prints headers
                        for (Product product : products) //prints product with capacity of 1 TB
                        {
                                if(product.getHardDiskCapacity().equals("1 TB"))
                      {
                         System.out.print(product);
                      }
                        }
                }
                break;

             default:
                System.out.println("Invalid selection. Please try again.");
                System.exit(hardDiskSelection);
          }
                }catch(InputMismatchException e)
                {
                        System.out.println("Invalid selection, please input an integer number.");
                        validSelection2=false;
                        sc.next();
                }
                catch(Exception e)
                {
                        System.out.println("Something is error.");

                }

                }while(!validSelection2);

            break;


        case 4: //(4) Display According to Internal Memory Capacity

                do

                {

                try //catch input mismatch error and other possible errors

                {

                validSelection=true;

                x=0; //to check if got product equals to 1 TB

                y=0; //to check if got product equals to 2 GB

                z=0; //to check if got product equals to 4 TB

                System.out.println("Select the Internal Memory Capacity.");

                menus.menu7(); //display internal memory capacity menu

            internalMemorySelection = sc.nextInt();

            while(internalMemorySelection < 1 || internalMemorySelection > 3) //validate selection
in range

            {

              System.out.println("Invalid selection. Please input between 1 and 3.");

                        System.out.println("Select the Internal Memory Capacity.");

              menus.menu7(); //display internal memory capacity menu

              internalMemorySelection = sc.nextInt();

            }
            switch(internalMemorySelection){
              case 1: //(1) 1 TB
                validSelection2=true;
                System.out.println("The records with internal memory equals to 1 TB");
                for(Product product : products)
                {
                  if(product.getInternalMemoryCapacity().equals("1 TB"))
                  {
                    x++;
                  }
                }
                if (x==0)
                {
                        System.out.println("No record with internal memory capacity of 1 TB.");
                }
                else if (x!=0)
                {
                        menus.menu(); //print headers
                        for (Product product : products) //print product with capacity of 1 TB
                        {
                              if(product.getInternalMemoryCapacity().equals("1 TB"))
                    {
                        System.out.print(product);
                    }
                }
            }
            break;

        case 2: //(2) 2 GB
           validSelection2=true;
            System.out.println("The records with internal memory equals to 2 GB");
            for(Product product : products)
            {
                if(product.getInternalMemoryCapacity().equals("2 GB"))
                {
                    y++;
                }

            }
            if (y==0)
            {
                System.out.println("No record with internal memory capacity of 2 GB.");
            }
            else if (y!=0)
            {
                menus.menu(); //print headers
                for (Product product : products) //print products with capacity of 2 GB
                {
```

```
                        if(product.getInternalMemoryCapacity().equals("2 GB"))
                {
                    System.out.print(product);
                }
                  }
            }
            break;


          case 3: //(3) 4 GB
            validSelection2=true;
            System.out.println("The records with internal memory equals to 4 GB");
            for(Product product : products)
            {
                if(product.getInternalMemoryCapacity().equals("4 GB"))
                {
                    z++;
                }
            }
            if (z==0)
            {
                        System.out.println("No record with internal memory capacity of 4 GB.");
            }
            else if (z!=0)
            {
                    menus.menu(); //print headers

                    for (Product product : products) //print product with capacity of 4 GB

                    {

                                if(product.getInternalMemoryCapacity().equals("4 GB"))

                {

                     System.out.print(product);

                }

                    }

        }

        break;


    default:

        System.out.println("Invalid selection. Please try again.");

        System.exit(internalMemorySelection);

}

        }catch(InputMismatchException e)

        {

                    System.out.println("Invalid selection, please input an integer number.");

                    validSelection2=false;

                    sc.next();

        }

        catch(Exception e)

        {

                    System.out.println("Something is error.");
        }

        }while(!validSelection2);

      break;


    case 5: //(5) Display According to Quantity

        boolean validQuantity = false;

        do

        {

        try //catch input mismatch error and other possible erros

        {

        x=0; //to check if there is product that quantity same with input quantity

        validSelection=true;

        int quant;

        System.out.println("Please enter the number of quantity.");

        System.out.print("Quantity: ");

        quant = sc.nextInt();

        while(quant < 0) //validate quantity should be positive

        {

            System.out.println("Invalid number of quantity. Please enter an positive integer.");

            System.out.println("Please enter the number of quantity.");

            System.out.print("Quantity: ");

        quant = sc.nextInt();

        }

        validQuantity = true;
        for(Product product : products)
{
    if(product.getQuantity()==(quant))
    {
        x++;
    }
}
        if (x==0)
        {
                    System.out.println("No record with quanity of "+quant+".");
        }
        else if (x!=0)
{
        menus.menu(); //print headers
        for (Product product : products) //print products with selected quantity
        {
                    if(product.getQuantity()==(quant))
        {
            System.out.print(product);
        }
        }
}
        }catch(InputMismatchException e)
        {
                    System.out.println("Invalid quantity, please input an integer number.");
                    validQuantity=false;
                    sc.next();
        }
        catch(Exception e)
        {
                    System.out.println("Something is error.");
        }
        }while(!validQuantity);
break;

case 6: //(6) Display All
        validSelection = true;
        menus.menu(); //print headers
        for (Product product : products) //print all products
        {
        System.out.print(product);
        }

case 7: //(7) Exit
        validSelection = true;
        break;
}
        }catch(InputMismatchException e)
```

```
        {

            System.out.println("Invalid selection, please input an integer number.");

            validSelection=false;

            sc.next();

        }

        catch(Exception e)

        {

            System.out.println("Something is error.");

        }

   }while(displaySelection!=7 || !validSelection); //will keep looping if user not choose to exit or invalid selection

    }

}
```

1.1.3 Menu.java

```
package Assignment1;

public class Menu {

public void menu() //Header

{

System.out.println("===================================================="

+ "====================================================================");

System.out.println("Product Code \tCountry \tProcessor\tHard Disk Capacity\t"

+ "Internal Memory Capacity\tQuantity");

System.out.println("===================================================="

+ "====================================================================");

}
```

```
public void menu1() //Product code table

{

System.out.println("\t\t Product Code Table");

System.out.println("========================================================
=====");

System.out.println("\tMeaning \t Characters \t Translation");

System.out.println("========================================================
=====");

System.out.println("Manufacturing Country \t\t M \t\t Malaysia");

System.out.println("\t\t\t\t J \t\t Japan");

System.out.println("\t\t\t\t A \t\t America");

System.out.println("========================================================
=====");

System.out.println("Type of Processor \t\t 5 \t\t Intel i5");

System.out.println("\t\t\t\t 7 \t\t Intel i7");

System.out.println("\t\t\t\t 9 \t\t Intel i9");

System.out.println("========================================================
=====");

System.out.println("Capacity of Hard Disk \t\t 320 \t\t 320 GB");

System.out.println("\t\t\t\t 500 \t\t 500 GB");

System.out.println("\t\t\t\t 1024 \t\t 1 TB");

System.out.println("========================================================
=====");

System.out.println("Capacity of Internal Memory \t 1024 \t\t 1 TB");

System.out.println("\t\t\t\t 2048 \t\t 2 GB");

System.out.println("\t\t\t\t 4096 \t\t 4 GB");
System.out.println("========================================================
=====");

System.out.println("The product code table is only for references before any updation or
deletion.\n");

}

public void menu2() //Main Menu

{

System.out.println("Please key in your selection.");

System.out.println("(1) Add New Products");

System.out.println("(2) Update Records");

System.out.println("(3) Delete Records");

System.out.println("(4) Display Records Based on Different Criteria");

System.out.println("(5) Display Product Code Table");

System.out.println("(6) Exit ");

System.out.print("Selection: ");

}

public void menu3() //(4) Display Records Based on Different Criteria

{

System.out.println("Select the criteria you wish to view products based on it.");

System.out.println("(1) Display According to Manufacturing Country");

System.out.println("(2) Display According to Processor Type");

System.out.println("(3) Display According to Hard Disk Capacity");

System.out.println("(4) Display According to Internal Memory Capacity");

System.out.println("(5) Display According to Quantity");

System.out.println("(6) Display All");
System.out.println("(7) Exit");
System.out.print("Selection: ");
}
public void menu4() //(1) Manufacturing Country
{
System.out.println("(1) Malaysia");
System.out.println("(2) Japan");
System.out.println("(3) America");
System.out.print("Selection: ");
}
public void menu5() //(2) Processor Type
{
System.out.println("(1) Intel i5");
System.out.println("(2) Intel i7");
System.out.println("(3) Intel i9");
System.out.print("Selection: ");
}
public void menu6() //(3) Hard Disk Capacity
{
System.out.println("(1) 320 GB");
System.out.println("(2) 500 GB");
System.out.println("(3) 1 TB");
System.out.print("Selection: ");
}
```

```
public void menu7() //(4) Internal Memory Capacity

{

System.out.println("(1) 1 TB ");

System.out.println("(2) 2 GB ");

System.out.println("(3) 4 GB ");

System.out.print("Selection: ");

}

public void menu8() //(2) Update Records

{

System.out.println("Select the criteria of the product you wish to update.");

System.out.println("(1) Manufacturing Country");

System.out.println("(2) Processor Type");

System.out.println("(3) Hard Disk Capacity");

System.out.println("(4) Internal Memory Capacity");

System.out.println("(5) Quantity");

System.out.println("(6) Exit");

System.out.print("Selection: ");

}

public void menu9() //Update all or update one by one

{

System.out.println("(1) Update All");

System.out.println("(2) Update One");

System.out.print("Selection: ");

}
```

```
public void menu10() //Yes or no menu

{

System.out.println("(1) Yes");

System.out.println("(2) No");

System.out.print("Selection: ");

}

public void menu11() //(3) Delete Records

{

System.out.println("(1) Delete Records");

System.out.println("(2) Exit");

System.out.print("Selection: ");

}

public void menu12() // (1) Add New Products

{

System.out.println("(1) Add Records");

System.out.println("(2) Exit");

System.out.print("Selection: ");

}

}
```

1.1.4 Product.java

```
package Assignment1;

public class Product {

//instance variables
```

```
private String productCode; //the code that identifies the product
private String country; //the country where the product is manufactured
private String processor; //the type of processor used in the product
private String hardDiskCapacity; // the capacity of the hard disk in the product
private String internalMemoryCapacity; //the capacity of the internal memory in the product
private int quantity; //the quantity of the product available
// Constructor
public Product(String productCode,String country, String processor,
String hardDiskCapacity, String internalMemoryCapacity,int quantity)
{
this.productCode = productCode;
this.country = country;
this.processor = processor;
this.hardDiskCapacity = hardDiskCapacity;
this.internalMemoryCapacity = internalMemoryCapacity;
this.quantity = quantity;
}
//Empty constructor for creating an object with default values
public Product()
{
}
//Getters and Setters for the instance variables
public String getProductCode()
{
```

```
return productCode;
}
public void setProductCode(String productCode)
{
this.productCode = productCode;
}
public String getCountry()
{
return country;
}
public void setCountry(String country)
{
this.country = country;
}
public String getProcessor()
{
return processor;
}
public void setProcessor(String processor)
{
this.processor = processor;
}
public String getHardDiskCapacity()
{
```

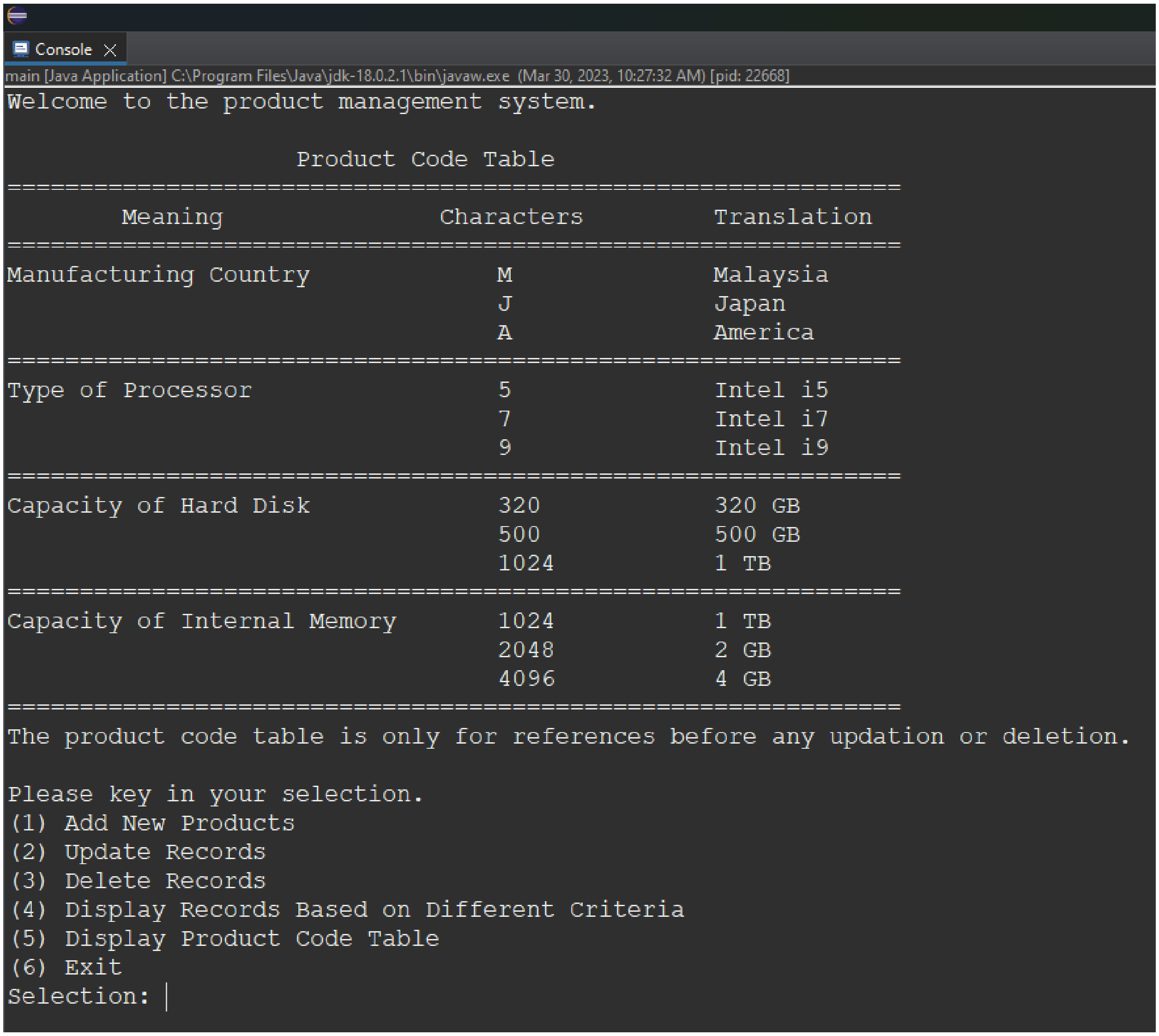
```
return hardDiskCapacity;
}
public void setHardDiskCapacity(String hardDiskCapacity)
{
this.hardDiskCapacity = hardDiskCapacity;
}
public String getInternalMemoryCapacity()
{
return internalMemoryCapacity;
}
public void setInternalMemoryCapacity(String internalMemoryCapacity)
{
this.internalMemoryCapacity = internalMemoryCapacity;
}
public int getQuantity()
{
return quantity;
}
public void setQuantity(int quantity)
{
this.quantity = quantity;
}
// Returns a formatted string representation of the object
public String toString()
```

```
{

return String.format("%-16s%-17s%-20s%-28s%-26s%d\n", productCode,country, processor,

hardDiskCapacity, internalMemoryCapacity, quantity);

}

}
```

1.2 Documentation

1.2.1 Normal Flow of Program

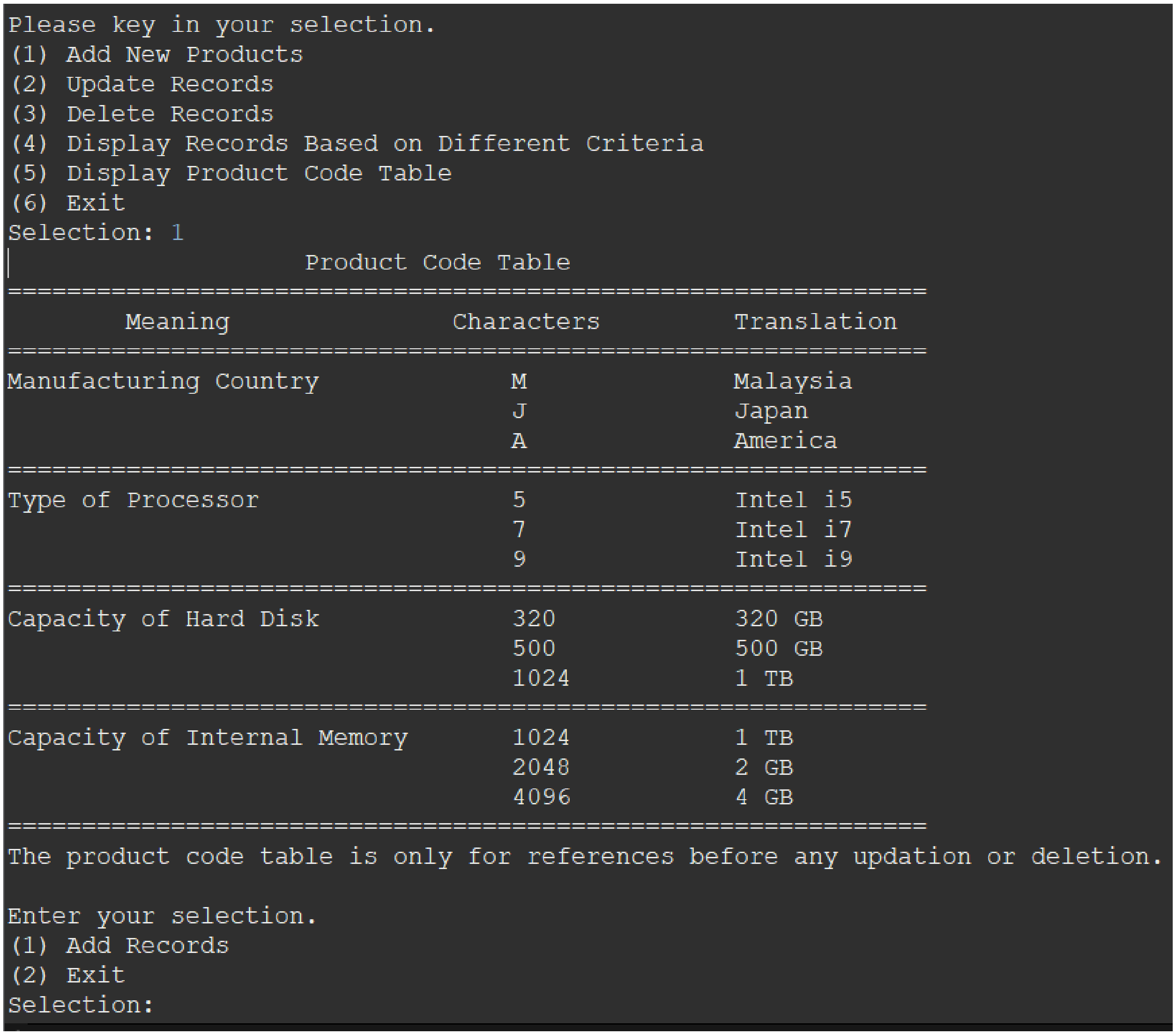
After running the Main.java file, the program first greets to the user by welcoming them to the system. The system then displays a product code table and mention that this is only for references before any modification on the product records. The system displays the main menu of the product management system. The menu consists of the selection of add new products, update records, delete records, display records based on different criteria, display product code table as well as the option to exit the program as shown in *Figure 1*. Users need to key in their selection by the number labelled before the option. To demonstrate all the functionality of the program, first, the user key in 1 for adding new product records to the system.



```
main [Java Application] C:\Program Files\Java\jdk-18.0.2.1\bin\javaw.exe (Mar 30, 2023, 10:27:32 AM) [pid: 22668]
Welcome to the product management system.

                    Product Code Table
==============================================================
        Meaning               Characters          Translation
==============================================================
Manufacturing Country              M              Malaysia
                                   J              Japan
                                   A              America
==============================================================
Type of Processor                  5              Intel i5
                                   7              Intel i7
                                   9              Intel i9
==============================================================
Capacity of Hard Disk              320            320 GB
                                   500            500 GB
                                   1024           1 TB
==============================================================
Capacity of Internal Memory        1024           1 TB
                                   2048           2 GB
                                   4096           4 GB
==============================================================
The product code table is only for references before any updation or deletion.

Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 
```

Figure 1 After running the Main.java file

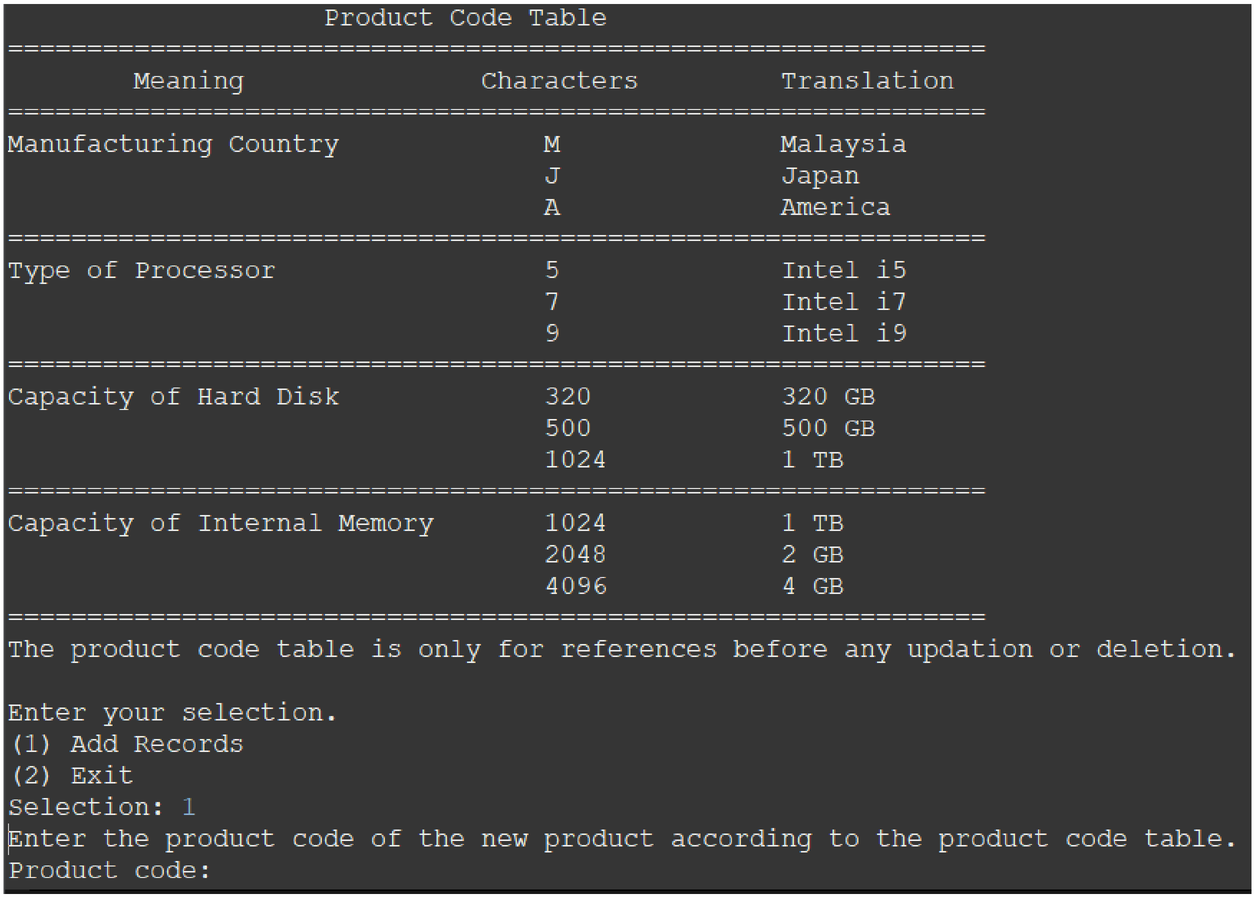
After the user input 1 for the main menu selection, the program displays the product code table again for the user to reference when key in the new product code. The program ask user to choose whether to add records or exit the add products menu as shown in *Figure 2*.



```
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 1
                    Product Code Table
==============================================================
        Meaning                Characters         Translation
==============================================================
Manufacturing Country             M              Malaysia
                                  J              Japan
                                  A              America
==============================================================
Type of Processor                 5              Intel i5
                                  7              Intel i7
                                  9              Intel i9
==============================================================
Capacity of Hard Disk             320            320 GB
                                  500            500 GB
                                  1024           1 TB
==============================================================
Capacity of Internal Memory       1024           1 TB
                                  2048           2 GB
                                  4096           4 GB
==============================================================
The product code table is only for references before any updation or deletion.

Enter your selection.
(1) Add Records
(2) Exit
Selection:
```

Figure 2 After selecting (1) Add New Products in main menu

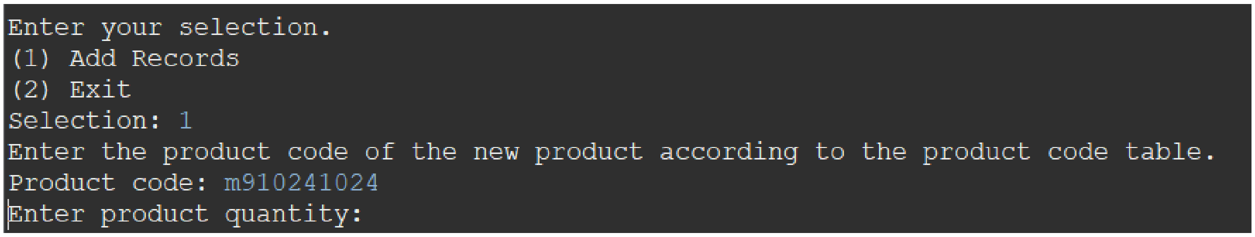
After user key in 1 to add records, the system then prompts for the user to enter the product code of the new product they intend to add into the system as shown in *Figure 3*.



```
                    Product Code Table
==============================================================
        Meaning                Characters         Translation
==============================================================
Manufacturing Country             M              Malaysia
                                  J              Japan
                                  A              America
==============================================================
Type of Processor                 5              Intel i5
                                  7              Intel i7
                                  9              Intel i9
==============================================================
Capacity of Hard Disk             320            320 GB
                                  500            500 GB
                                  1024           1 TB
==============================================================
Capacity of Internal Memory       1024           1 TB
                                  2048           2 GB
                                  4096           4 GB
==============================================================
The product code table is only for references before any updation or deletion.

Enter your selection.
(1) Add Records
(2) Exit
Selection: 1
Enter the product code of the new product according to the product code table.
Product code:
```

Figure 3 After selecting (1) Add Records in add product menu

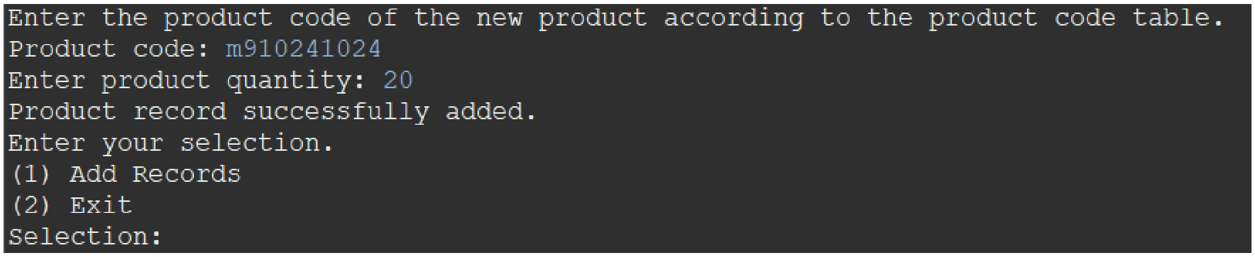
After user input a valid product code, the system will prompt for the quantity of the new product as shown in *Figure 4*.



```
Enter your selection.
(1) Add Records
(2) Exit
Selection: 1
Enter the product code of the new product according to the product code table.
Product code: m910241024
Enter product quantity:
```

Figure 4 After inputting valid product code

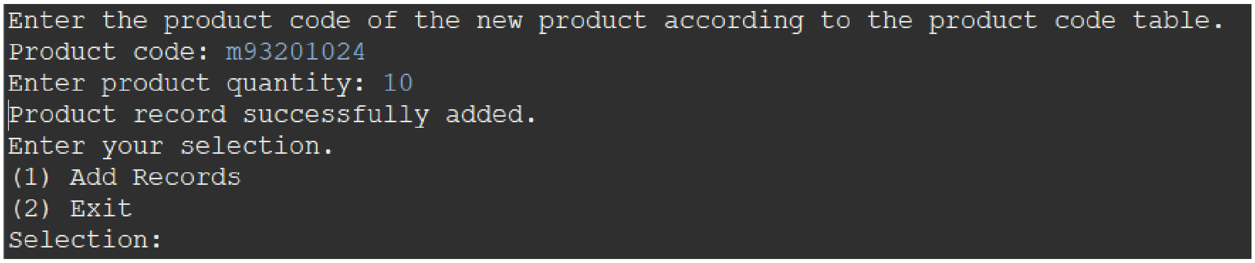
After user input a valid quantity for the new product, the system will then prompt that the new product record has successfully added into the system as shown in *Figure 5*. The system will then return to the add product menu and ask for the user to select whether exit the add product menu or continue to add new product record.



```
Enter the product code of the new product according to the product code table.
Product code: m910241024
Enter product quantity: 20
Product record successfully added.
Enter your selection.
(1) Add Records
(2) Exit
Selection:
```

Figure 5 Product record successfully added

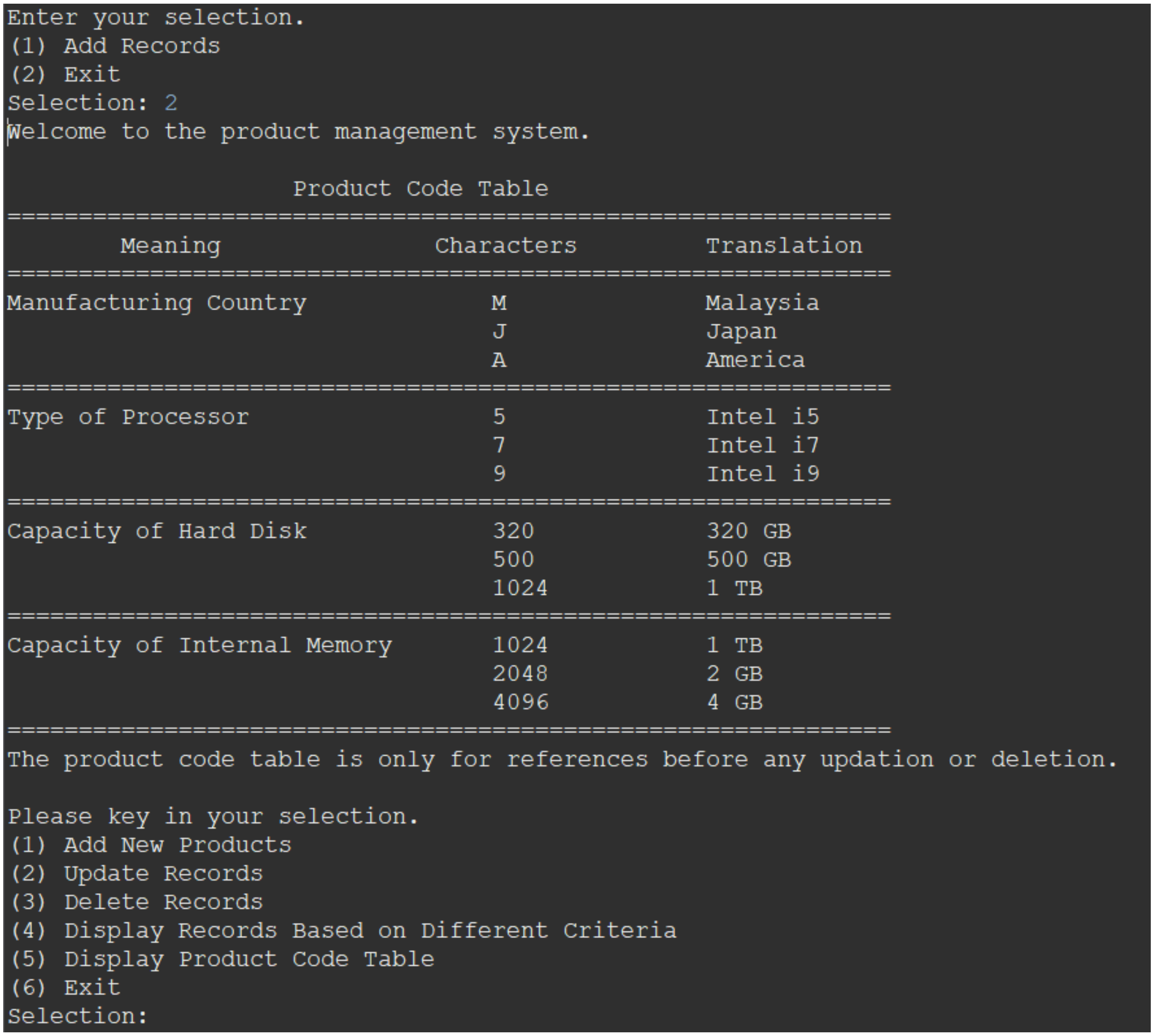
Then, the user continues to add another product record with the product code of a valid product code and quantity, the system will prompt again that the new product record has successfully added as shown in *Figure 6*.



```
Enter the product code of the new product according to the product code table.
Product code: m93201024
Enter product quantity: 10
Product record successfully added.
Enter your selection.
(1) Add Records
(2) Exit
Selection:
```

Figure 6 Another product record successfully added

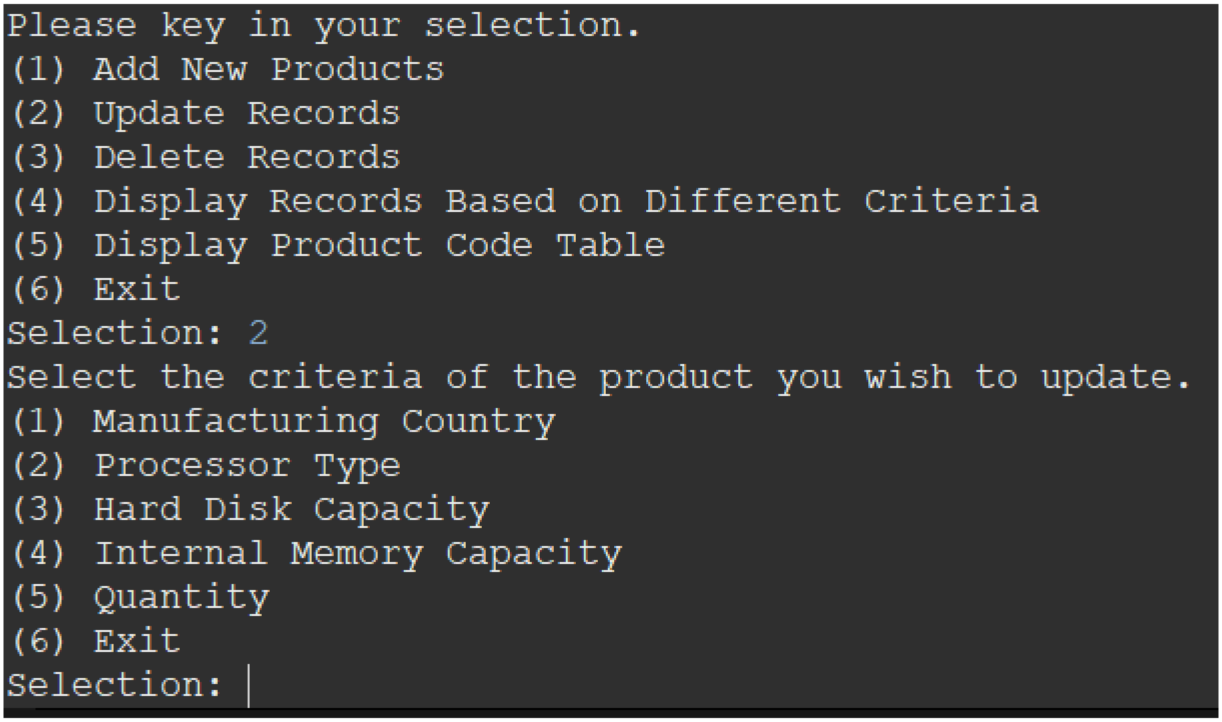
After the user input 2 to exit the add product menu, the system will then go back to the main menu of the product management system and ask for the user to input the selection for the main menu as shown in *Figure 7*.



```
Enter your selection.
(1) Add Records
(2) Exit
Selection: 2
Welcome to the product management system.

                    Product Code Table
==============================================================
        Meaning               Characters         Translation
==============================================================
Manufacturing Country             M              Malaysia
                                  J              Japan
                                  A              America
==============================================================
Type of Processor                 5              Intel i5
                                  7              Intel i7
                                  9              Intel i9
==============================================================
Capacity of Hard Disk             320            320 GB
                                  500            500 GB
                                  1024           1 TB
==============================================================
Capacity of Internal Memory       1024           1 TB
                                  2048           2 GB
                                  4096           4 GB
==============================================================
The product code table is only for references before any updation or deletion.

Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection:
```

Figure 7 Exit add new products menu

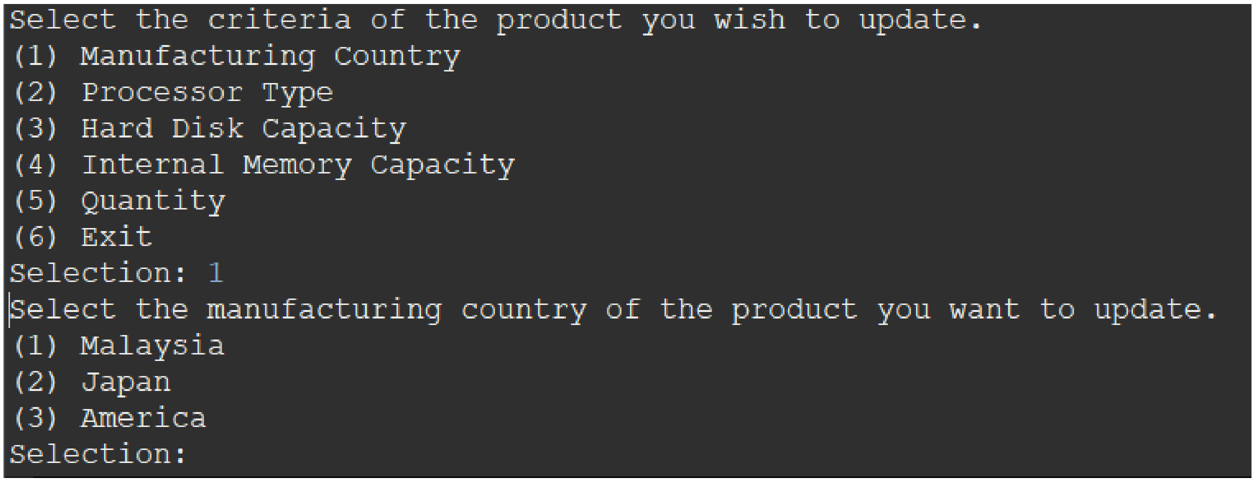
Then, after user input 2 to update records, the system will then display the menu for update records and ask user to key in the selection as shown in *Figure 8*.



```
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 2
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection:
```

Figure 8 display (2) Update Records menu

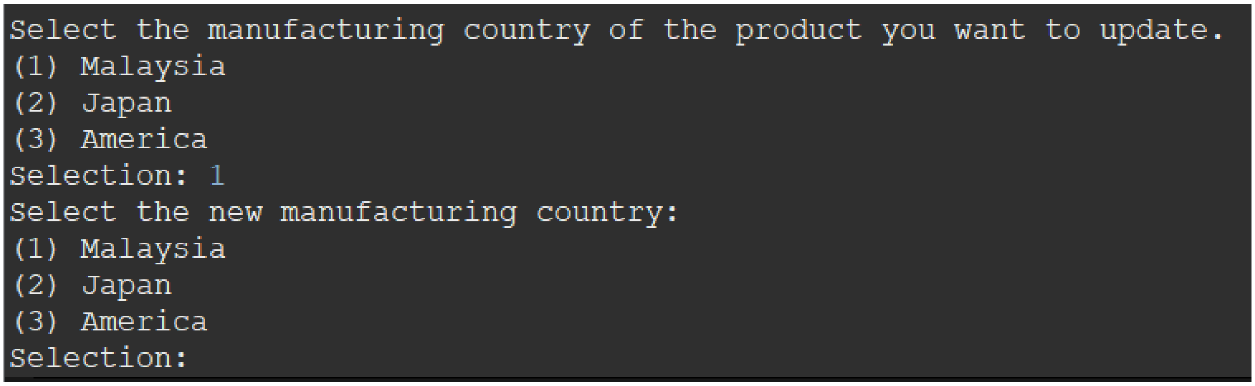
After user key in the selection of criteria to update, the system will then display the menu for the particular criteria and ask user to key in selection as shown in *Figure 9*.



```
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection: 1
Select the manufacturing country of the product you want to update.
(1) Malaysia
(2) Japan
(3) America
Selection:
```

Figure 9 Update (1) Manufacturing Country menu

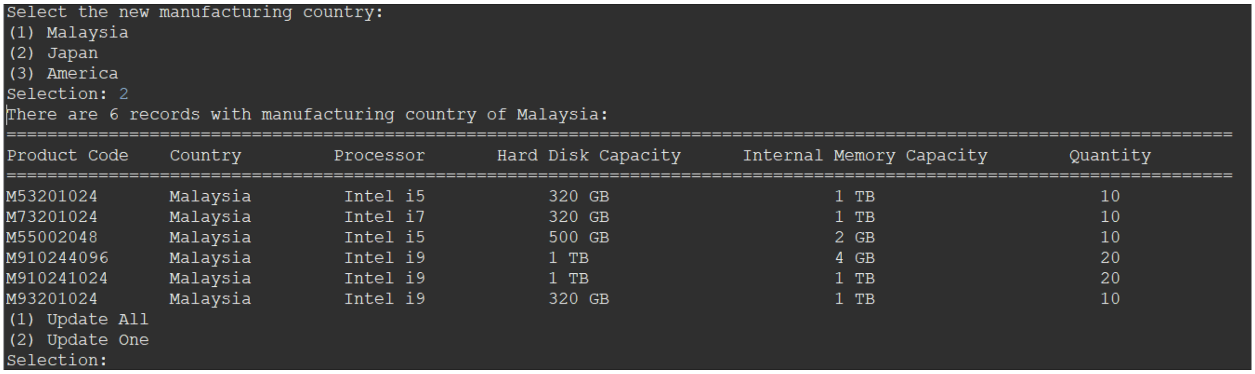
After user key in the selection for the country of the product that need to be updated, the system will then prompt the country menu again for the user to choose the new country to be update as and let user key in the selection as shown in *Figure 10*.



```
Select the manufacturing country of the product you want to update.
(1) Malaysia
(2) Japan
(3) America
Selection: 1
Select the new manufacturing country:
(1) Malaysia
(2) Japan
(3) America
Selection:
```

Figure 10 Country menu for new manufacturing country to update as

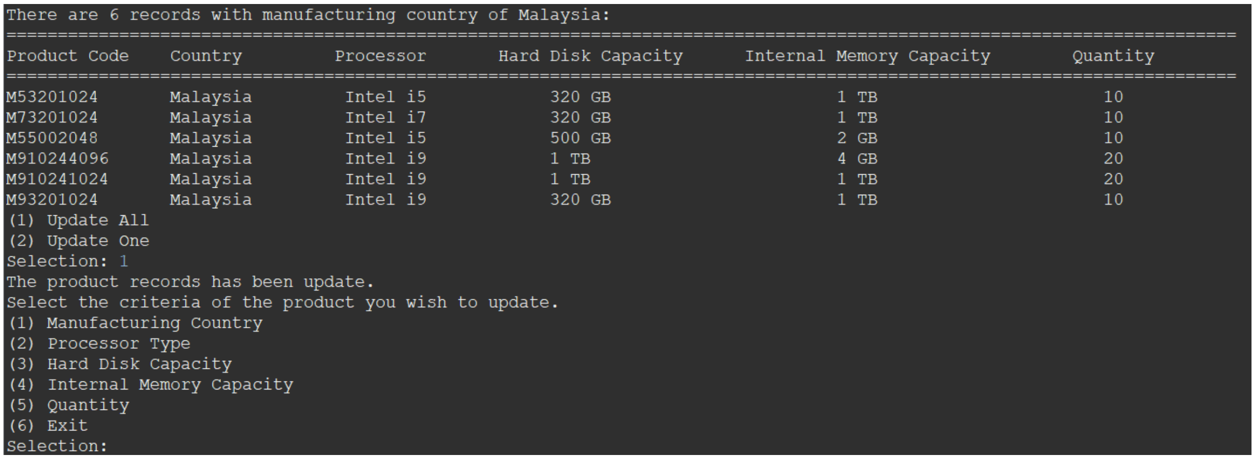
After the user select the new manufacturing country, the system will then prompt out the number of records that match to the manufacturing country of the record that need to be updated. Then, the system will display the list of the product records that match to the manufacturing country of the record that need to be updated. After that, the system asks the user to choose whether to update all records or update only one record as shown in *Figure 11*.



```
Select the new manufacturing country:
(1) Malaysia
(2) Japan
(3) America
Selection: 2
There are 6 records with manufacturing country of Malaysia:
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Malaysia         Intel i5            320 GB                        1 TB                      10
M73201024       Malaysia         Intel i7            320 GB                        1 TB                      10
M55002048       Malaysia         Intel i5            500 GB                        2 GB                      10
M910244096      Malaysia         Intel i9            1 TB                          4 GB                      20
M910241024      Malaysia         Intel i9            1 TB                          1 TB                      20
M93201024       Malaysia         Intel i9            320 GB                        1 TB                      10
(1) Update All
(2) Update One
Selection:
```

Figure 11 After user select new manufacturing country to update as

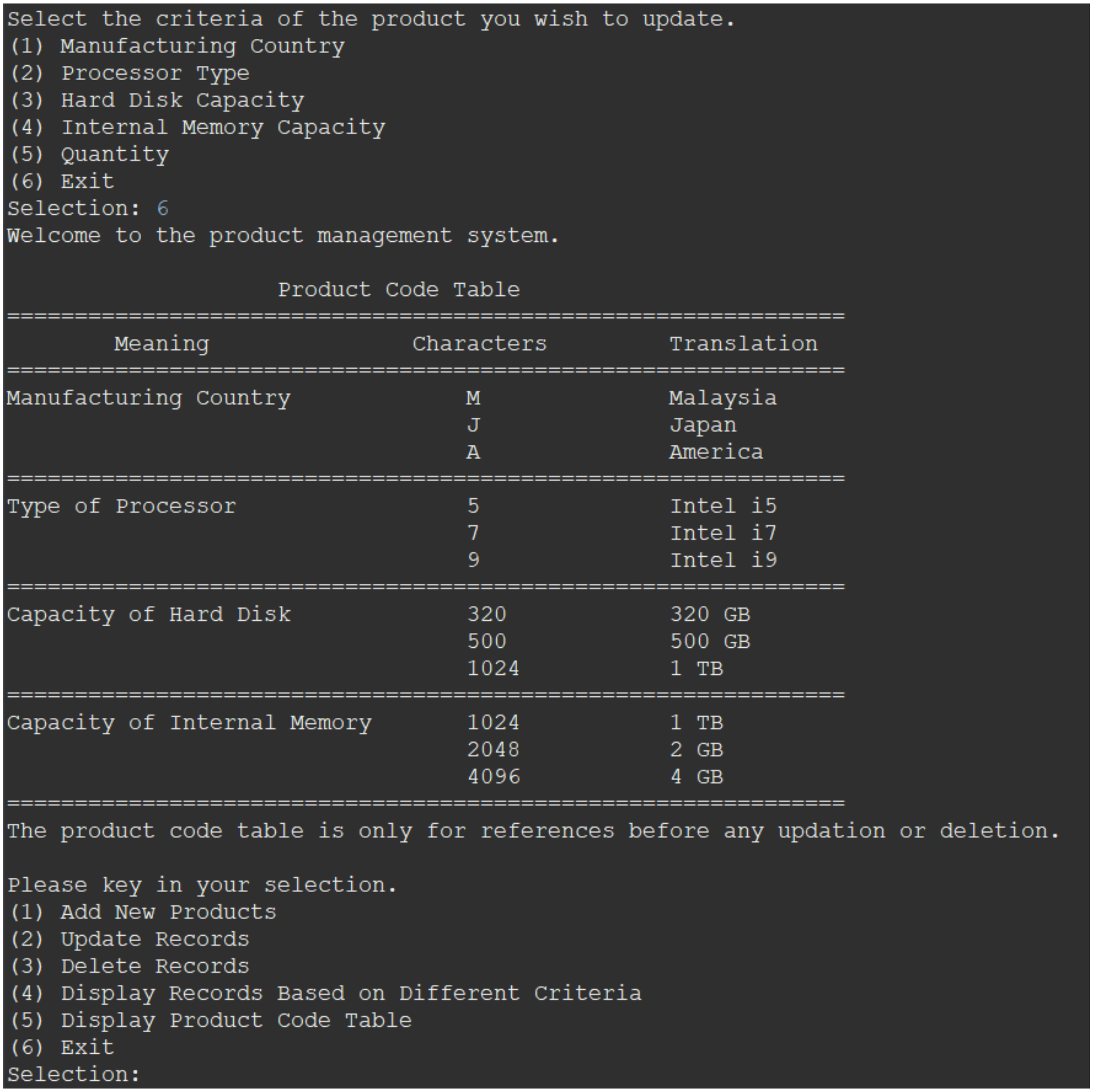
If the user chooses to update all, the system will then prompt that the product records has been updated and return to the update product records menu as shown in *Figure 12*.



```
There are 6 records with manufacturing country of Malaysia:
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Malaysia         Intel i5            320 GB                        1 TB                      10
M73201024       Malaysia         Intel i7            320 GB                        1 TB                      10
M55002048       Malaysia         Intel i5            500 GB                        2 GB                      10
M910244096      Malaysia         Intel i9            1 TB                          4 GB                      20
M910241024      Malaysia         Intel i9            1 TB                          1 TB                      20
M93201024       Malaysia         Intel i9            320 GB                        1 TB                      10
(1) Update All
(2) Update One
Selection: 1
The product records has been update.
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection:
```

Figure 12 User choose to update all

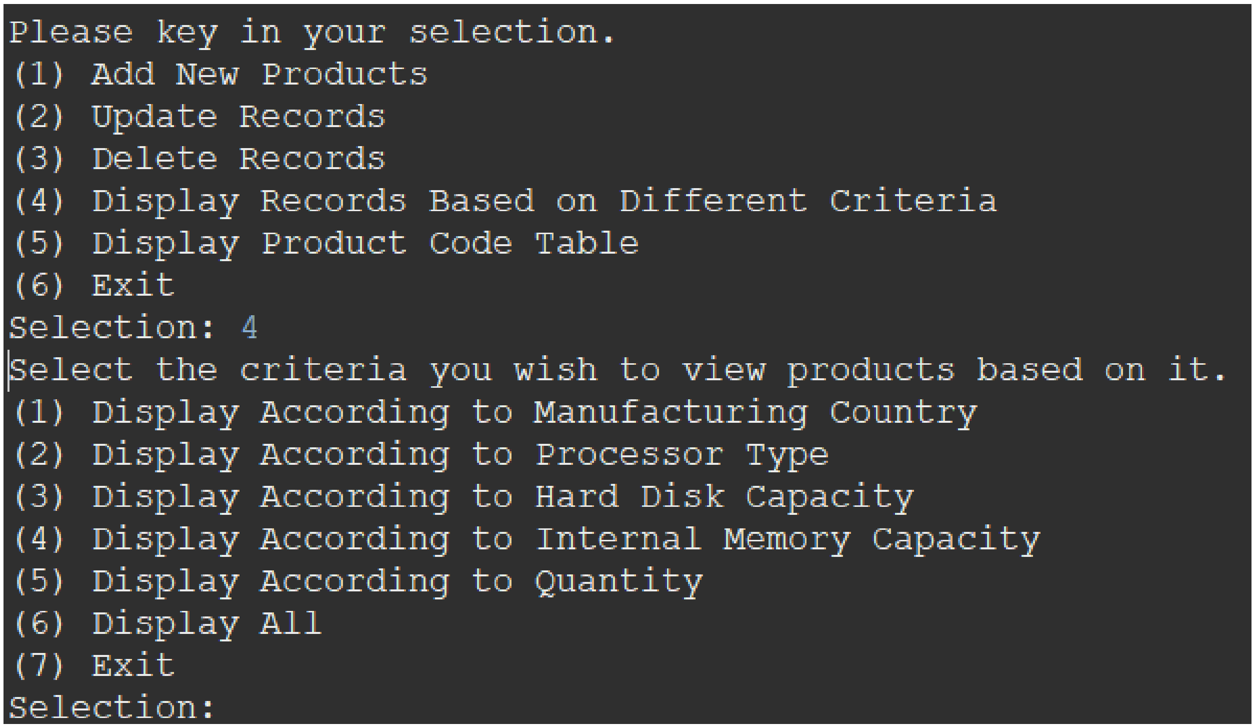
After that, user key in 6 to exit the program as shown in *Figure 13*.



```
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection: 6
Welcome to the product management system.

                    Product Code Table
==============================================================
        Meaning              Characters         Translation
==============================================================
Manufacturing Country            M              Malaysia
                                 J              Japan
                                 A              America
==============================================================
Type of Processor                5              Intel i5
                                 7              Intel i7
                                 9              Intel i9
==============================================================
Capacity of Hard Disk            320            320 GB
                                 500            500 GB
                                 1024           1 TB
==============================================================
Capacity of Internal Memory      1024           1 TB
                                 2048           2 GB
                                 4096           4 GB
==============================================================
The product code table is only for references before any updation or deletion.

Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection:
```

Figure 13 User choose to exit the program

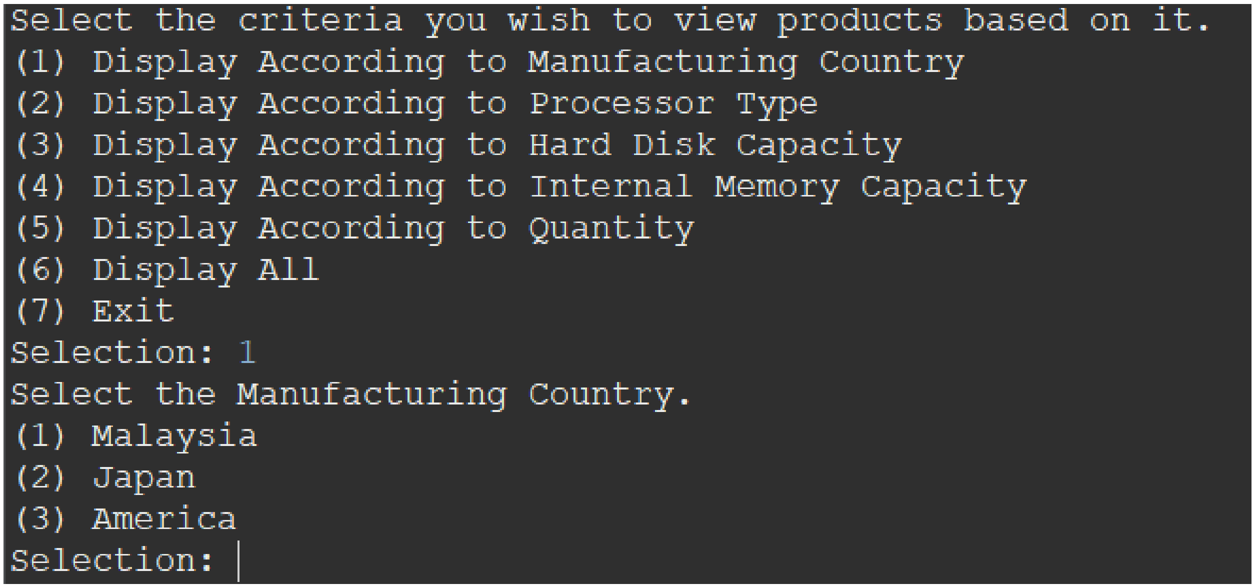
Then, the user intends to check if the records has been updated successfully and key in 4 to display product records. The system will then display the display menu as shown in *Figure 14*.



```
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 4
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

Figure 14 The system will then display the display menu

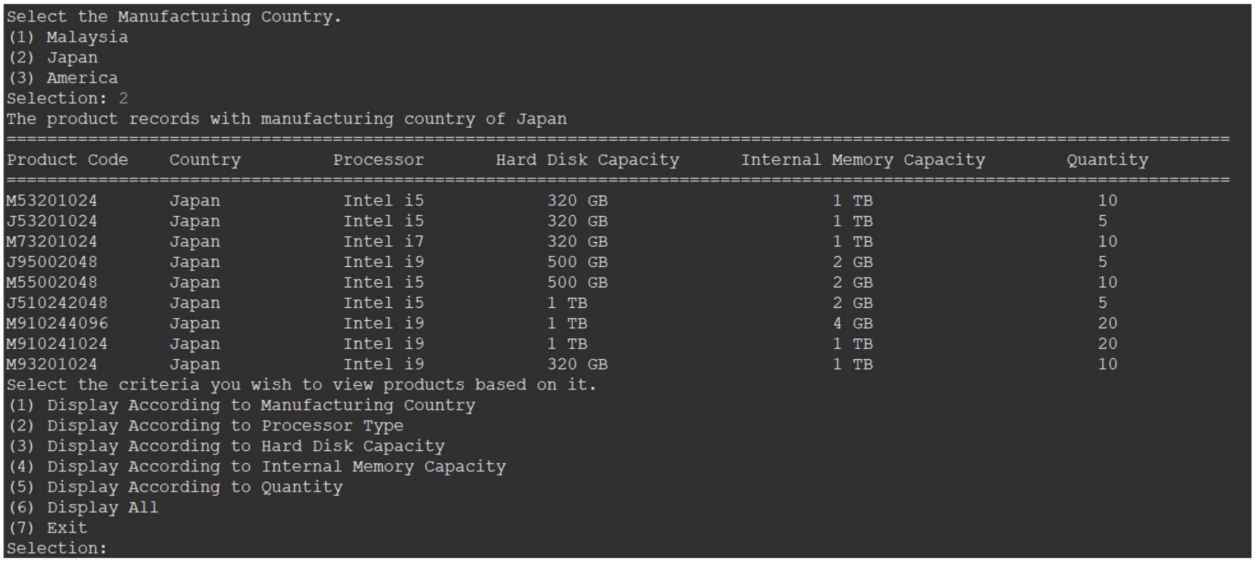
After the user key in 1 to display according to manufacturing country, the system will then display the country menu for the user to select as shown in *Figure 15*.



```
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection: 1
Select the Manufacturing Country.
(1) Malaysia
(2) Japan
(3) America
Selection:
```

Figure 15 After selecting the criteria to display according to

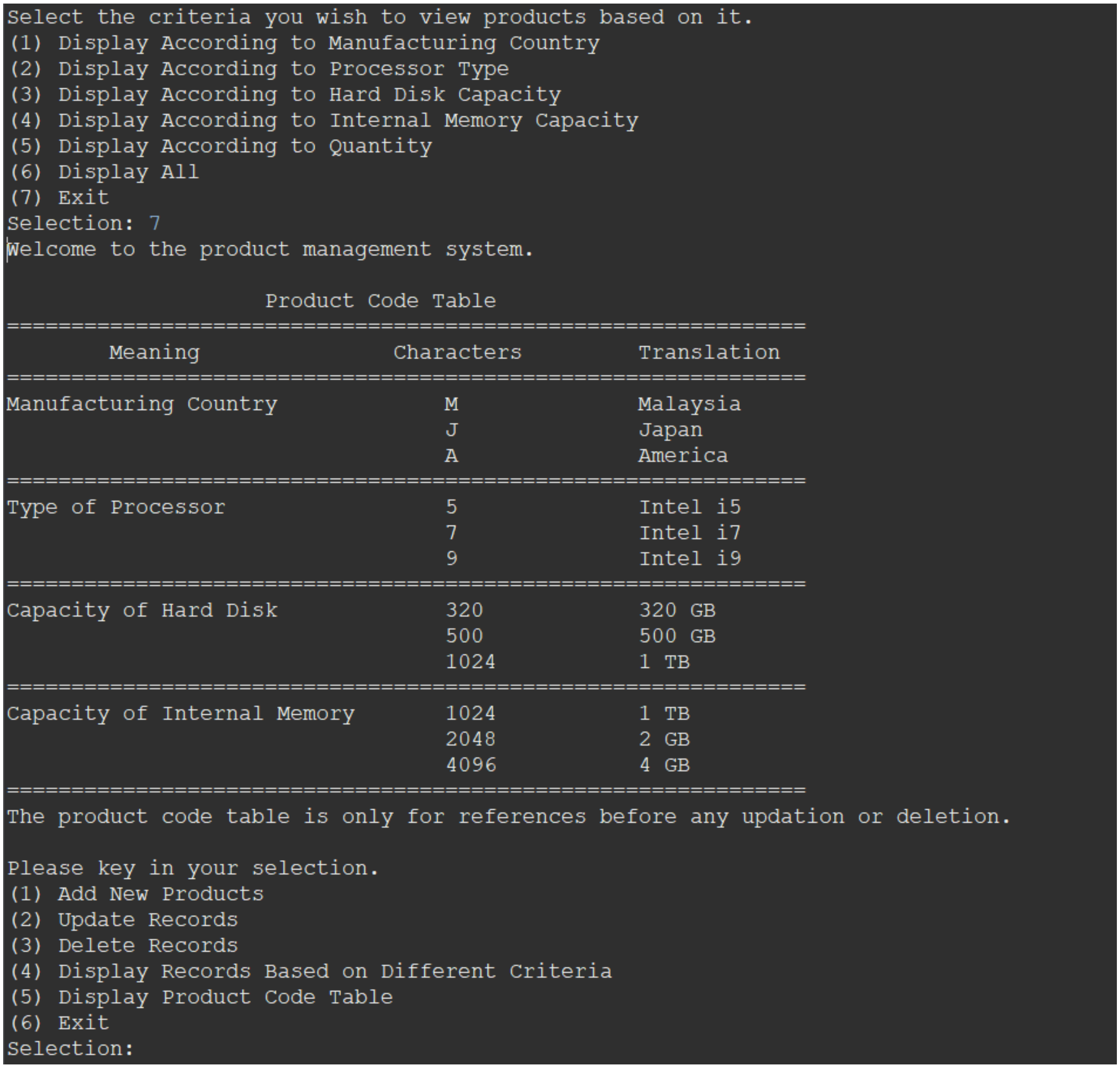
After user key in the country to display, the system will then display the product records list that its manufacturing country match to the selected country which is Japan in this case. After that, the system will return to the display menu and ask for selection from the user as shown in *Figure 16*.



```
Select the Manufacturing Country.
(1) Malaysia
(2) Japan
(3) America
Selection: 2
The product records with manufacturing country of Japan
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Japan            Intel i5           320 GB                      1 TB                      10
J53201024       Japan            Intel i5           320 GB                      1 TB                      5
M73201024       Japan            Intel i7           320 GB                      1 TB                      10
J95002048       Japan            Intel i9           500 GB                      2 GB                      5
M55002048       Japan            Intel i5           500 GB                      2 GB                      10
J510242048      Japan            Intel i5           1 TB                        2 GB                      5
M910244096      Japan            Intel i9           1 TB                        4 GB                      20
M910241024      Japan            Intel i9           1 TB                        1 TB                      20
M93201024       Japan            Intel i9           320 GB                      1 TB                      10
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

Figure 16 Display product records that country equals to Japan

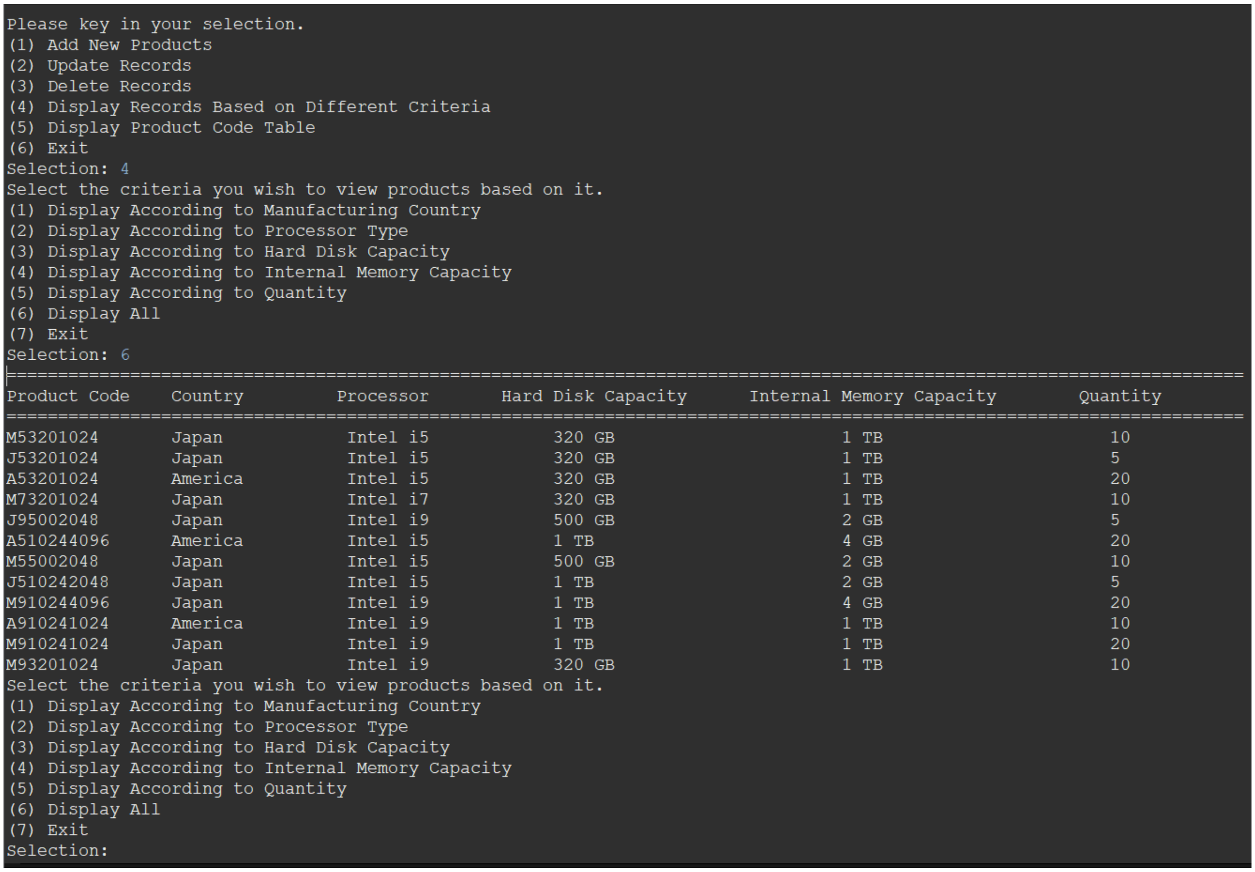
User can see that, the country of product code chose to update just now has changed from Malaysia to Japan (eg: M53201024). The user then enters 7 to exit the display menu and return to the main menu of the product management system as shown in *Figure 17*.



```
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection: 7
Welcome to the product management system.

                    Product Code Table
==============================================================
        Meaning               Characters         Translation
==============================================================
Manufacturing Country             M              Malaysia
                                  J              Japan
                                  A              America
==============================================================
Type of Processor                 5              Intel i5
                                  7              Intel i7
                                  9              Intel i9
==============================================================
Capacity of Hard Disk             320            320 GB
                                  500            500 GB
                                  1024           1 TB
==============================================================
Capacity of Internal Memory       1024           1 TB
                                  2048           2 GB
                                  4096           4 GB
==============================================================
The product code table is only for references before any updation or deletion.

Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection:
```

Figure 17 User enter 7 to exit the display product menu

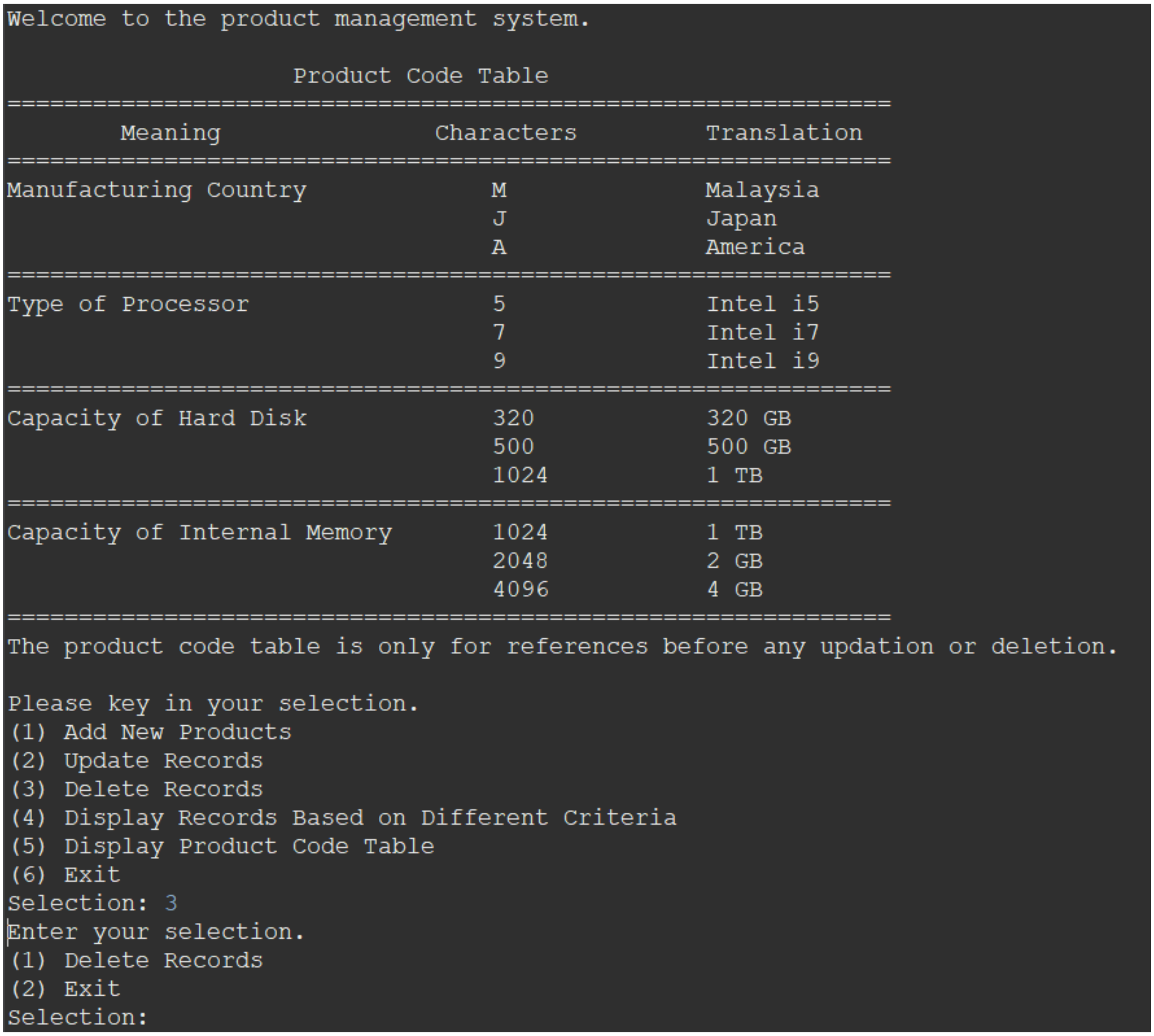
After that, user choose to display all records to check if the product code added in the beginning is added into the system. User key in 6 in the display menu to display all products as shown in *Figure 18*.



```
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 4
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection: 6
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Japan            Intel i5           320 GB                      1 TB                      10
J53201024       Japan            Intel i5           320 GB                      1 TB                      5
A53201024       America          Intel i5           320 GB                      1 TB                      20
M73201024       Japan            Intel i7           320 GB                      1 TB                      10
J95002048       Japan            Intel i9           500 GB                      2 GB                      5
A510244096      America          Intel i5           1 TB                        4 GB                      20
M55002048       Japan            Intel i5           500 GB                      2 GB                      10
J510242048      Japan            Intel i5           1 TB                        2 GB                      5
M910244096      Japan            Intel i9           1 TB                        4 GB                      20
A910241024      America          Intel i9           1 TB                        1 TB                      10
M910241024      Japan            Intel i9           1 TB                        1 TB                      20
M93201024       Japan            Intel i9           320 GB                      1 TB                      10
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

Figure 18 The user choose to display all product records

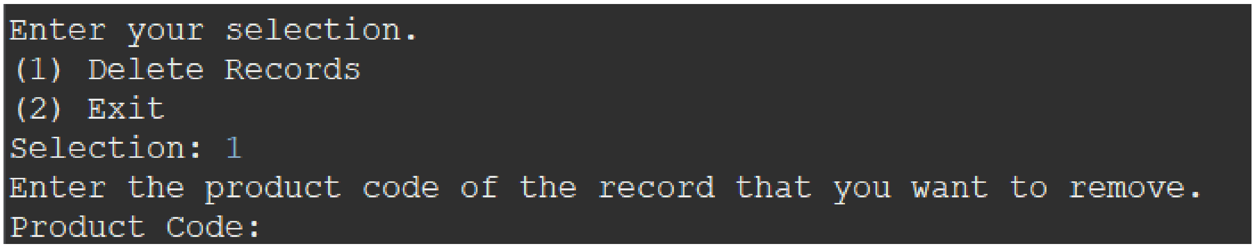
As shown in the product list, M910241024 and M93201024 has successfully added into the system, and as the manufacturing country of products with Malaysia as manufacturing country has updated as Japan, so the country of these two products is shown as Japan. Since the product has successfully added, the user intends to try the delete function and hence key in 7 to exit the display menu. Then, the user key in 3 in the main menu to delete records. The system then prompts the delete products menu to let user to choose whether to delete records or exit the delete records menu as shown in *Figure 19*.



```
Welcome to the product management system.

                    Product Code Table
==============================================================
        Meaning               Characters        Translation
==============================================================
Manufacturing Country             M              Malaysia
                                  J              Japan
                                  A              America
==============================================================
Type of Processor                 5              Intel i5
                                  7              Intel i7
                                  9              Intel i9
==============================================================
Capacity of Hard Disk             320            320 GB
                                  500            500 GB
                                  1024           1 TB
==============================================================
Capacity of Internal Memory       1024           1 TB
                                  2048           2 GB
                                  4096           4 GB
==============================================================
The product code table is only for references before any updation or deletion.

Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 3
Enter your selection.
(1) Delete Records
(2) Exit
Selection:
```

Figure 19 User choose to delete records in main menu

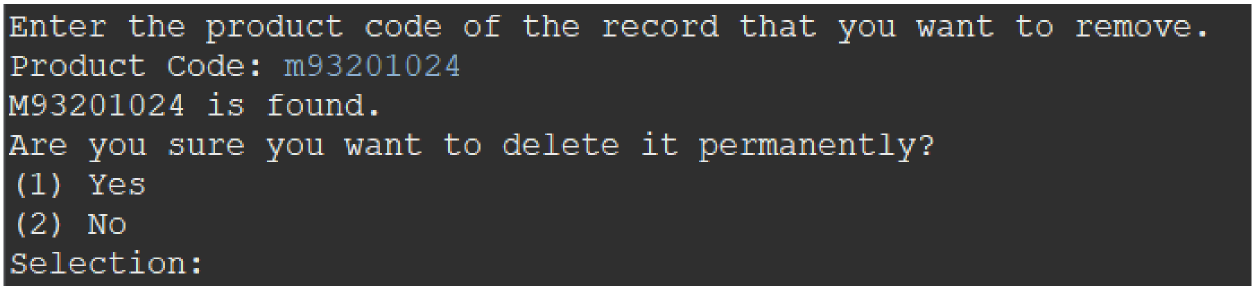
After user input 1 to delete records, the system prompted to let the user key in the product code of the product that need to be deleted as shown in *Figure 20*.



```
Enter your selection.
(1) Delete Records
(2) Exit
Selection: 1
Enter the product code of the record that you want to remove.
Product Code:
```

Figure 20 System ask for the product code to delete

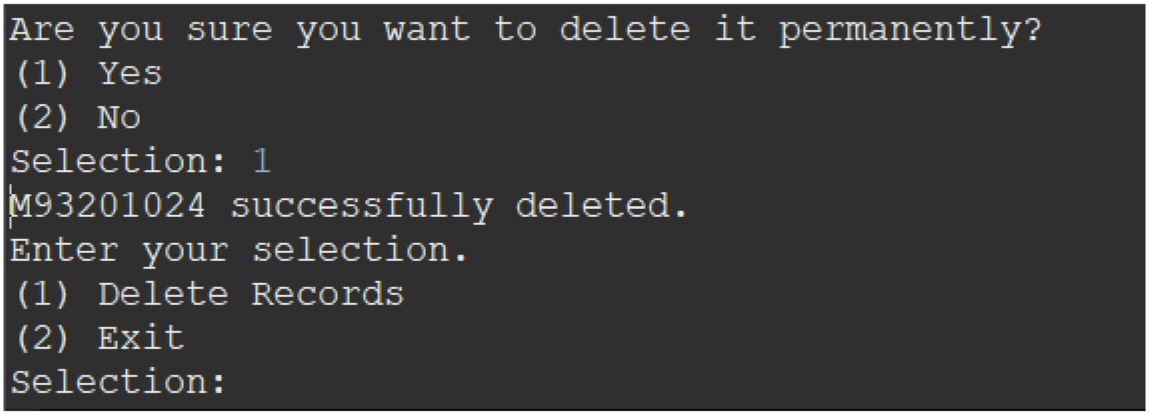
After the user key in the valid product code which the product records is in the system, the system will ask for the confirmation of the user to delete that particular record as shown in *Figure 21*.



```
Enter the product code of the record that you want to remove.
Product Code: m93201024
M93201024 is found.
Are you sure you want to delete it permanently?
(1) Yes
(2) No
Selection:
```

Figure 21 Confirmation of product record deletion

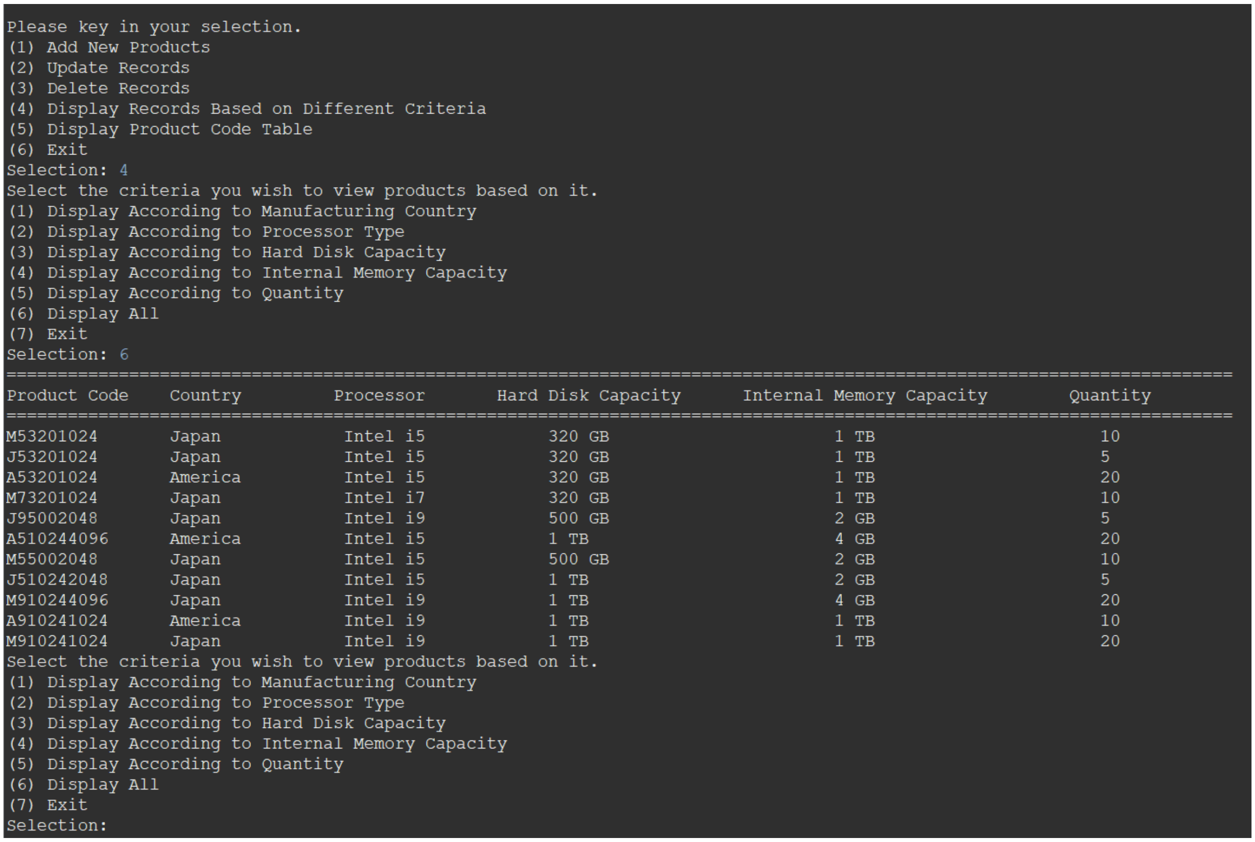
After user key in 1 to delete the product permanently, the system will prompt that the product code input by the user just now has been successfully deleted. After that, the system will prompt again the selection to let the user to choose whether continue delete other product or exit the delete product menu as shown in *Figure 22*.



```
Are you sure you want to delete it permanently?
(1) Yes
(2) No
Selection: 1
M93201024 successfully deleted.
Enter your selection.
(1) Delete Records
(2) Exit
Selection:
```

Figure 22 System ask for selection after product deletion

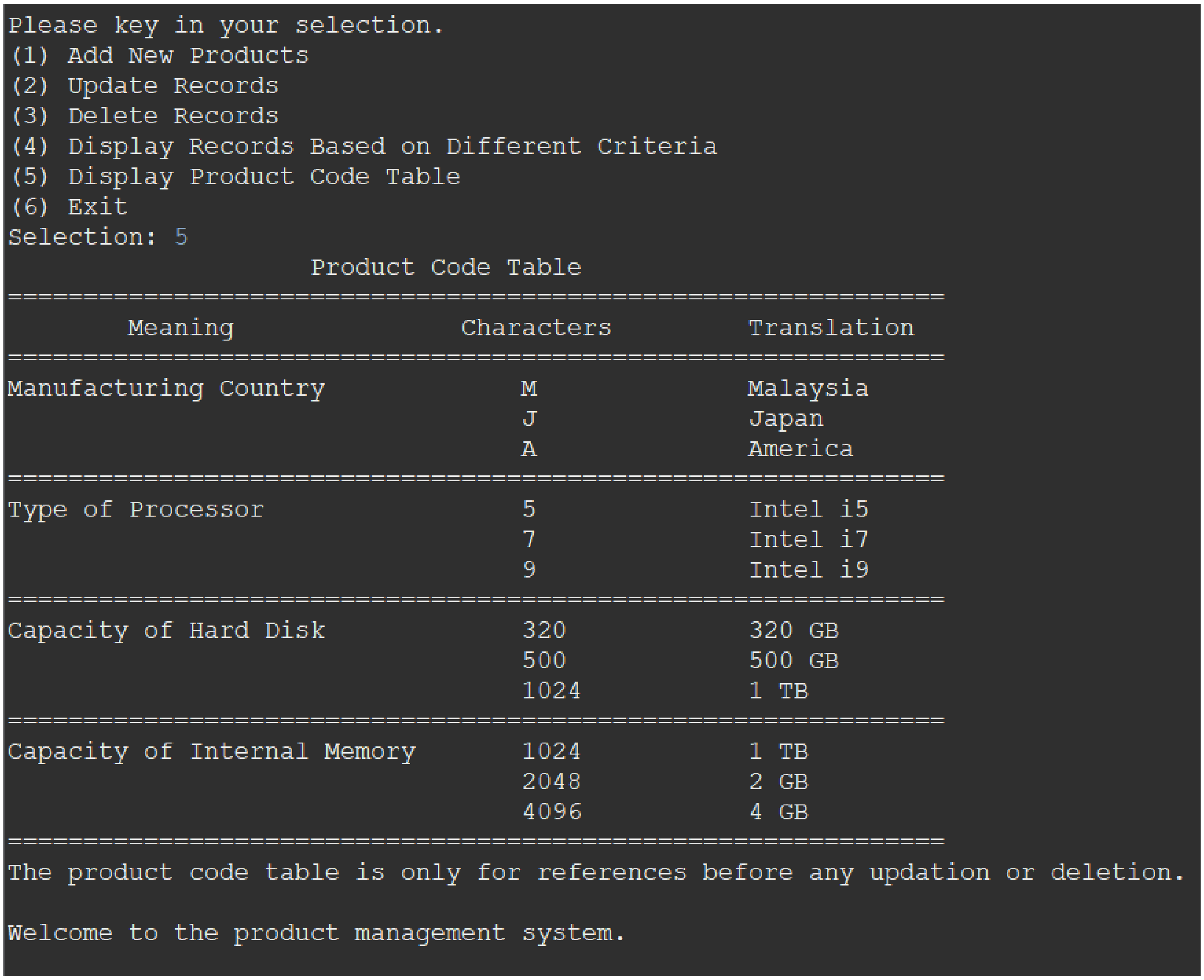
After that, the user intend to check whether the product selected to delete just now (M93201024) has really been deleted, hence key in 4 to enter display product menu. Then, the user key in 6 to show all product records as shown in *Figure 23*.



```
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 4
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection: 6
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Japan            Intel i5            320 GB                      1 TB                      10
J53201024       Japan            Intel i5            320 GB                      1 TB                      5
A53201024       America          Intel i5            320 GB                      1 TB                      20
M73201024       Japan            Intel i7            320 GB                      1 TB                      10
J95002048       Japan            Intel i9            500 GB                      2 GB                      5
A510244096      America          Intel i5            1 TB                        4 GB                      20
M55002048       Japan            Intel i5            500 GB                      2 GB                      10
J510242048      Japan            Intel i5            1 TB                        2 GB                      5
M910244096      Japan            Intel i9            1 TB                        4 GB                      20
A910241024      America          Intel i9            1 TB                        1 TB                      10
M910241024      Japan            Intel i9            1 TB                        1 TB                      20
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

Figure 23 Check for the deletion of product M93201024

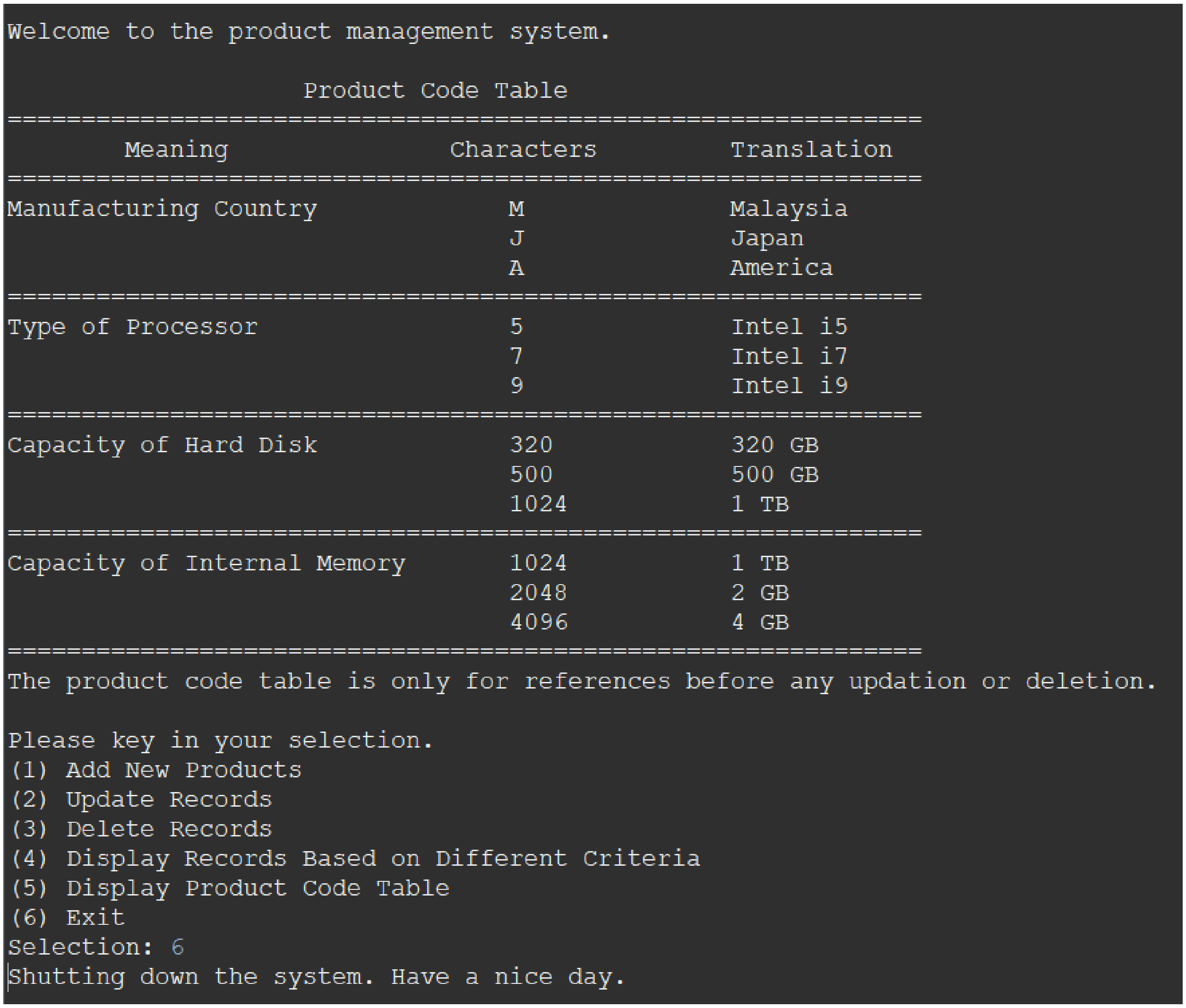
User can see that, the product record M93201024 has successfully deleted as display all records does not show the product record M93201024. Then, the user exits the delete product menu and intend to test the remaining feature. The user key in 5 to display the product code table. The system then prompt out the product code table with the notes that, the table is only for reference before any modification (updation, deletion) on the product records as shown in *Figure 24*.



```
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 5
                    Product Code Table
==============================================================
        Meaning               Characters        Translation
==============================================================
Manufacturing Country             M             Malaysia
                                  J             Japan
                                  A             America
==============================================================
Type of Processor                 5             Intel i5
                                  7             Intel i7
                                  9             Intel i9
==============================================================
Capacity of Hard Disk             320           320 GB
                                  500           500 GB
                                  1024          1 TB
==============================================================
Capacity of Internal Memory       1024          1 TB
                                  2048          2 GB
                                  4096          4 GB
==============================================================
The product code table is only for references before any updation or deletion.

Welcome to the product management system.
```

Figure 24 (5) Display Product Code Table

After the user tested all of the features, the user decided to shut down the system. Hence, the user key in 6 to exit the program. Then, the system prompted a message that the system is shutting down, and greets the user have a nice day. The program then eventually ends as shown in *Figure 25*.



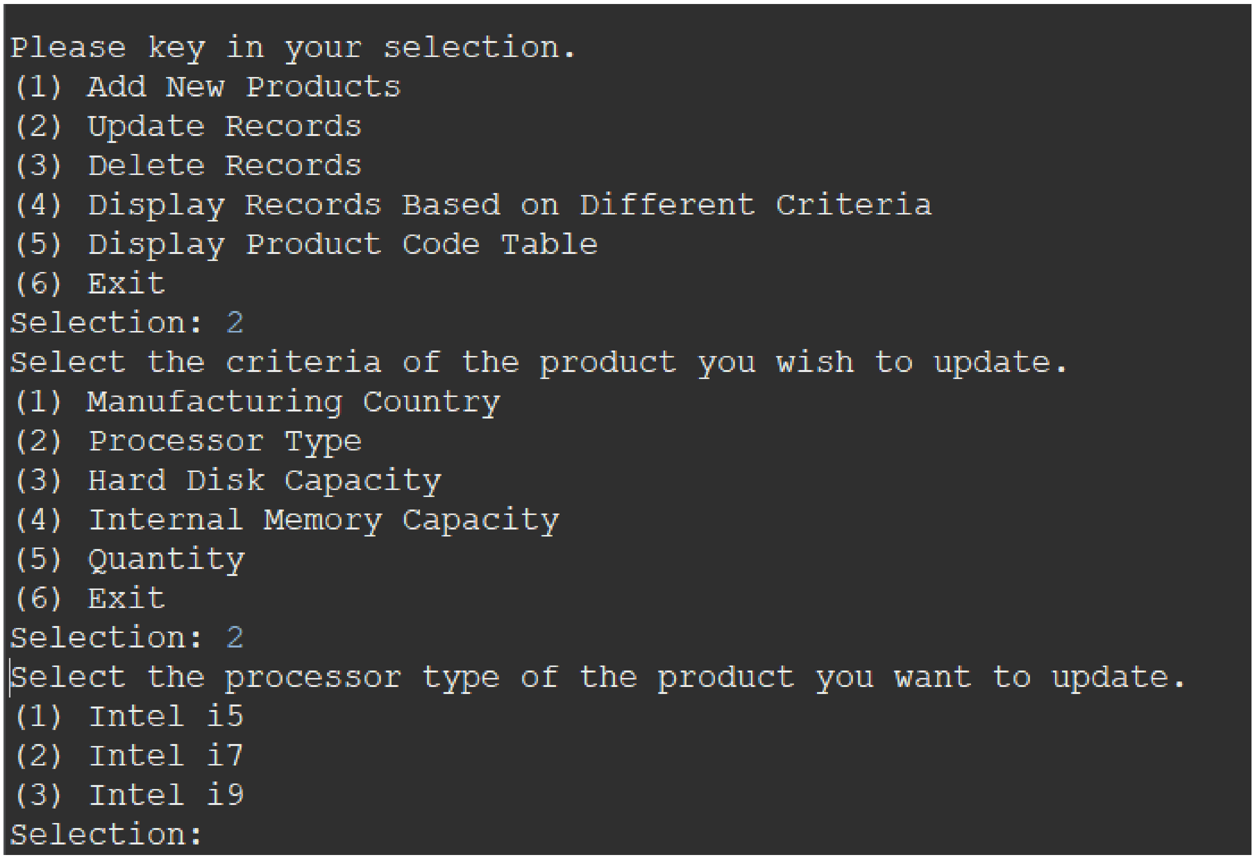
```
Welcome to the product management system.

                    Product Code Table
==============================================================
        Meaning               Characters         Translation
==============================================================
Manufacturing Country             M              Malaysia
                                  J              Japan
                                  A              America
==============================================================
Type of Processor                 5              Intel i5
                                  7              Intel i7
                                  9              Intel i9
==============================================================
Capacity of Hard Disk             320            320 GB
                                  500            500 GB
                                  1024           1 TB
==============================================================
Capacity of Internal Memory       1024           1 TB
                                  2048           2 GB
                                  4096           4 GB
==============================================================
The product code table is only for references before any updation or deletion.

Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 6
Shutting down the system. Have a nice day.
```

Figure 25 End of the Program

1.2.2 Alternative Case

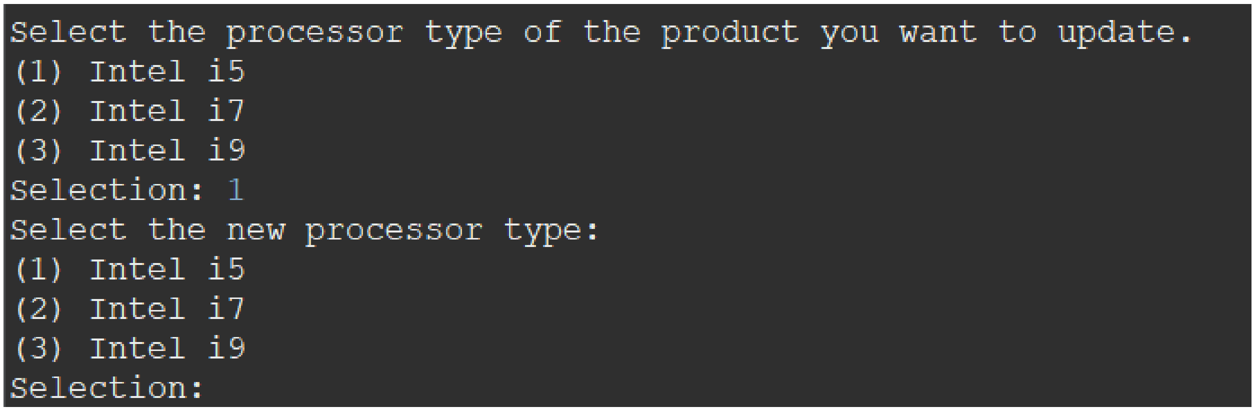
The alternative case is used to show the other possible output of the program which does not demonstrate in **1.2.1 Normal Flow of Program**.

(2) Update Records



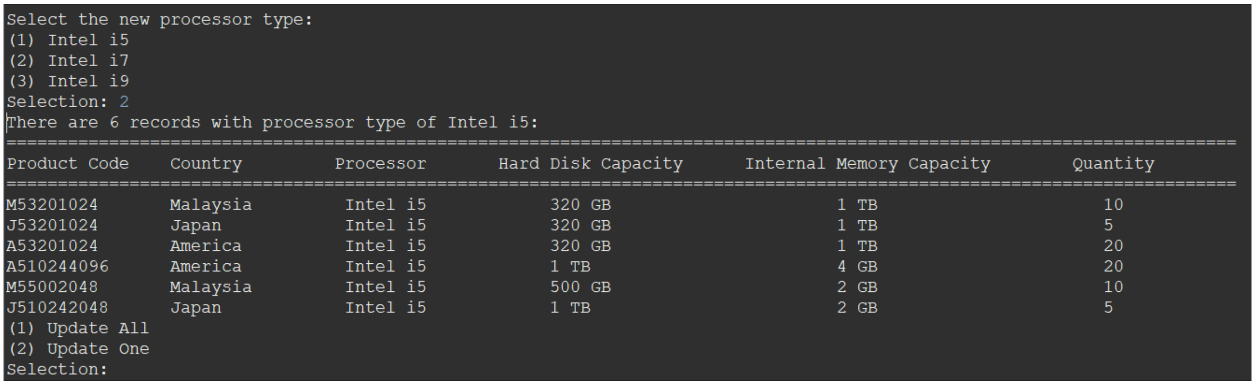
```
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 2
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection: 2
Select the processor type of the product you want to update.
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection:
```

Figure 26 - 1 Update (2) Processor Type



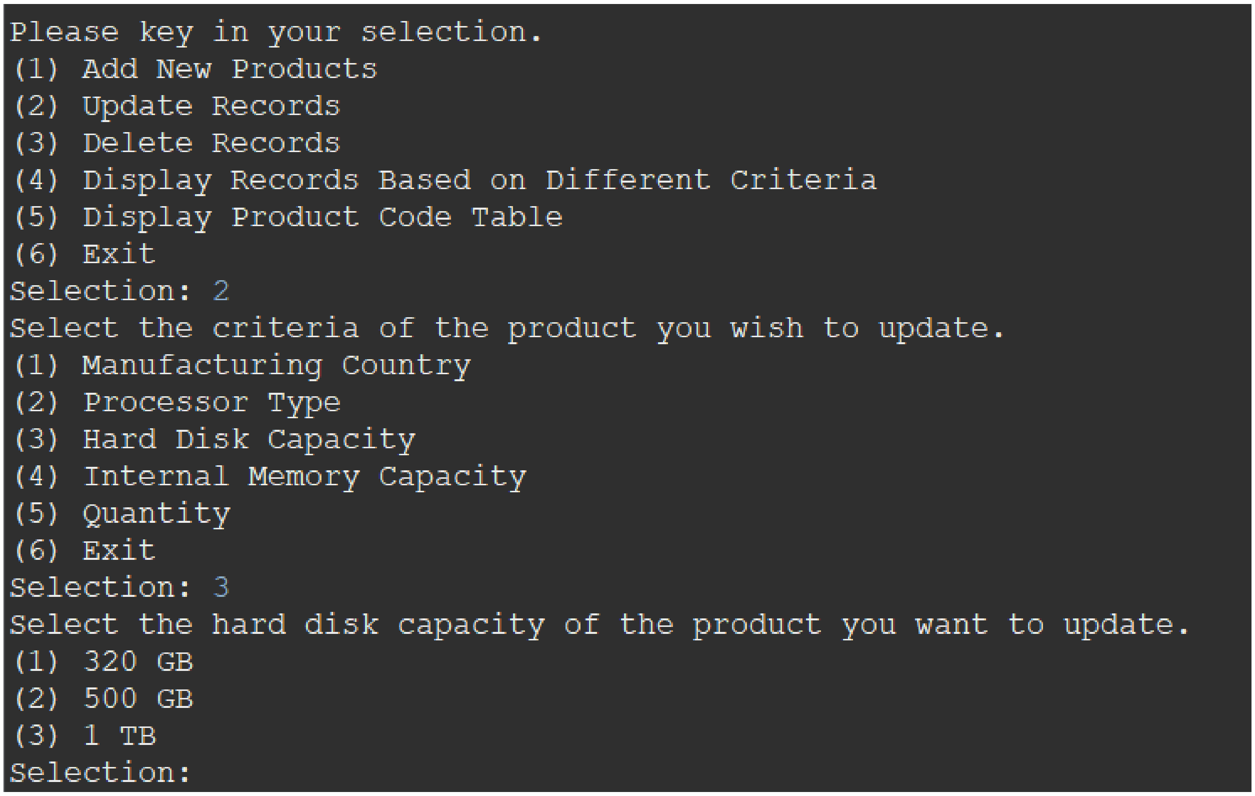
```
Select the processor type of the product you want to update.
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection: 1
Select the new processor type:
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection:
```

Figure 26 - 2 Update (2) Processor Type



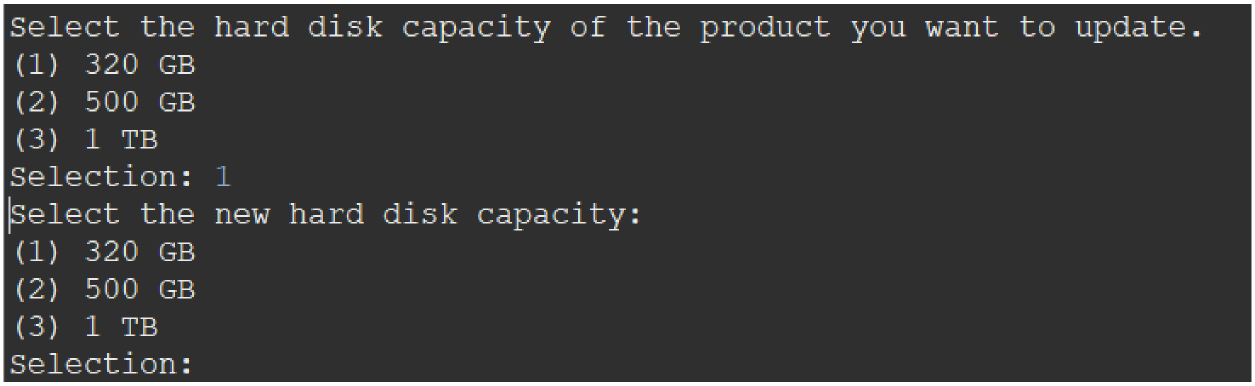
```
Select the new processor type:
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection: 2
There are 6 records with processor type of Intel i5:
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Malaysia         Intel i5           320 GB                      1 TB                      10
J53201024       Japan            Intel i5           320 GB                      1 TB                      5
A53201024       America          Intel i5           320 GB                      1 TB                      20
A510244096      America          Intel i5           1 TB                        4 GB                      20
M55002048       Malaysia         Intel i5           500 GB                      2 GB                      10
J510242048      Japan            Intel i5           1 TB                        2 GB                      5
(1) Update All
(2) Update One
Selection:
```

Figure 26 - 3 Update (2) Processor Type



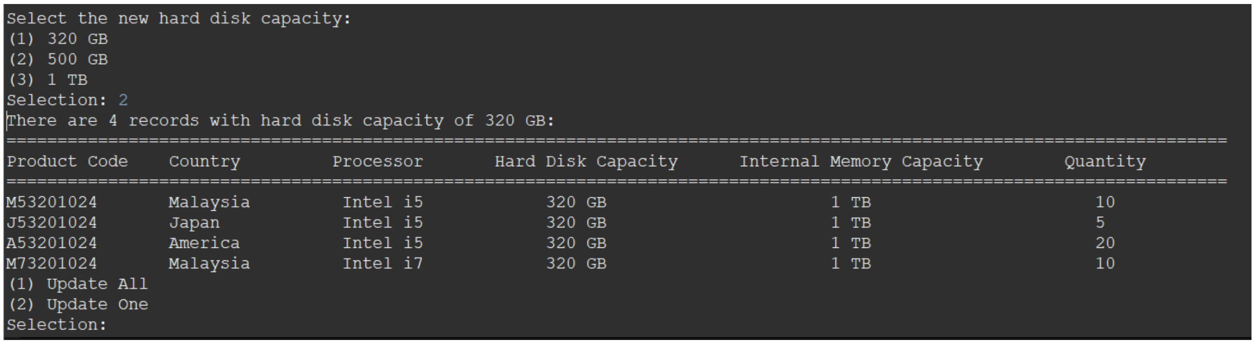
```
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 2
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection: 3
Select the hard disk capacity of the product you want to update.
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection:
```

Figure 27 – 1 Update (3) Hard Disk Capacity



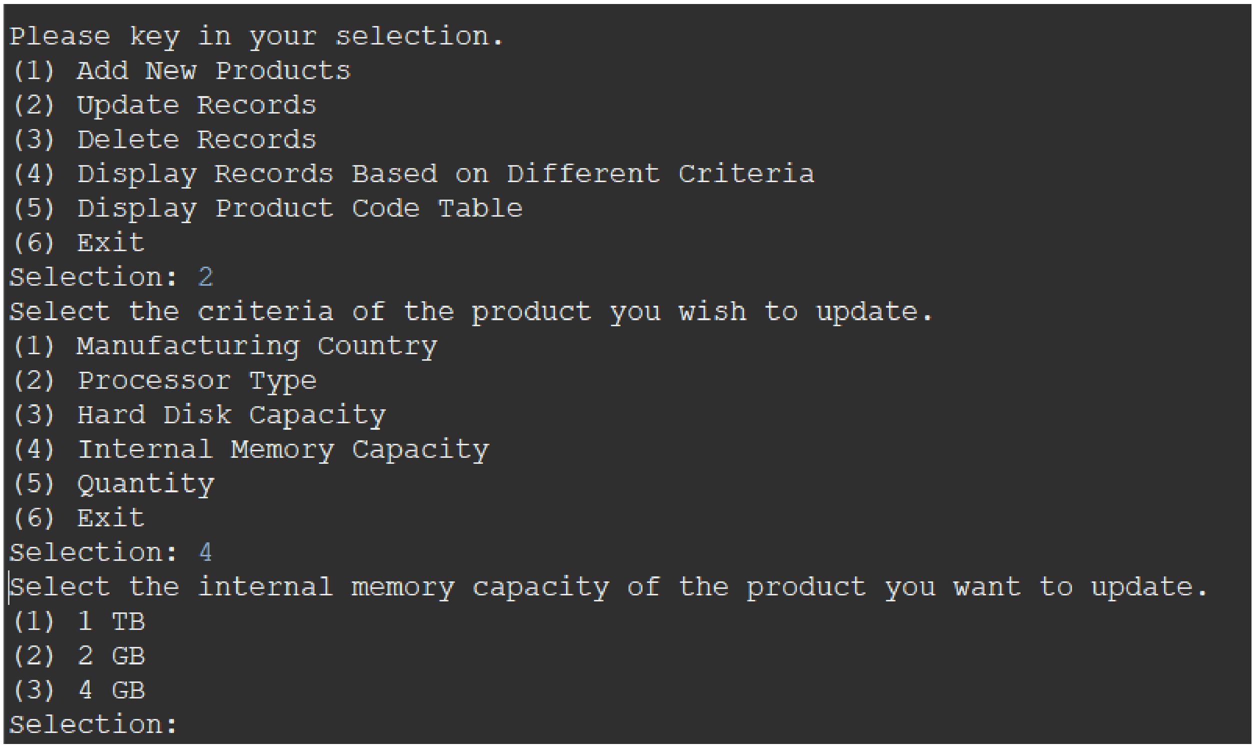
```
Select the hard disk capacity of the product you want to update.
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection: 1
Select the new hard disk capacity:
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection:
```

Figure 27 – 2 Update (3) Hard Disk Capacity



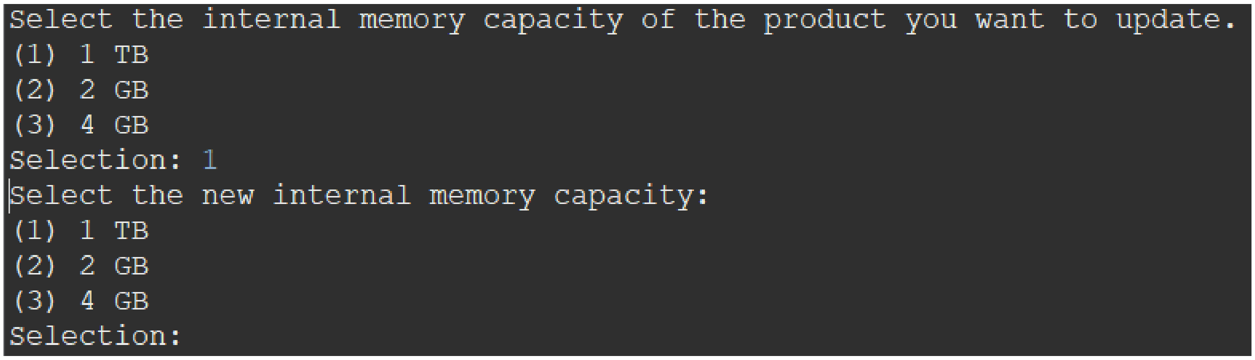
```
Select the new hard disk capacity:
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection: 2
There are 4 records with hard disk capacity of 320 GB:
============================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
============================================================================================================================
M53201024       Malaysia         Intel i5           320 GB                         1 TB                       10
J53201024       Japan            Intel i5           320 GB                         1 TB                       5
A53201024       America          Intel i5           320 GB                         1 TB                       20
M73201024       Malaysia         Intel i7           320 GB                         1 TB                       10
(1) Update All
(2) Update One
Selection:
```

Figure 27 – 3 Update (3) Hard Disk Capacity



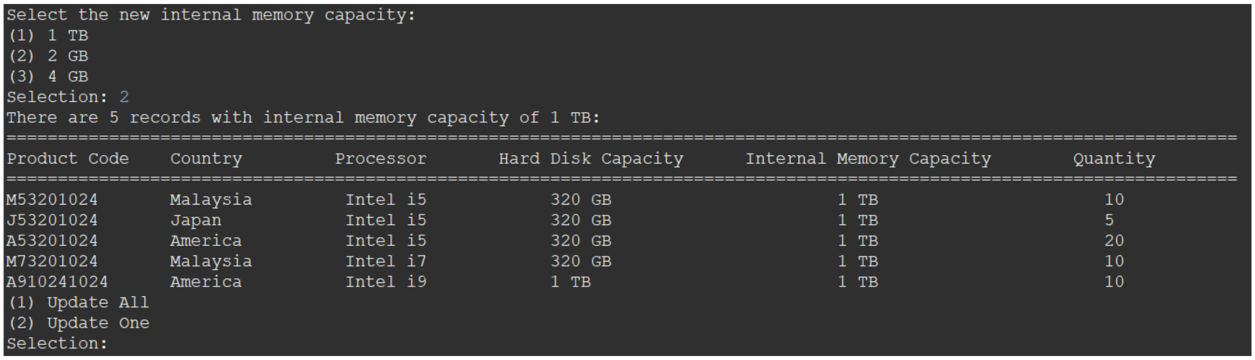
```
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 2
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection: 4
Select the internal memory capacity of the product you want to update.
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection:
```

Figure 28 - 1 Update (4) Internal Memory Capacity



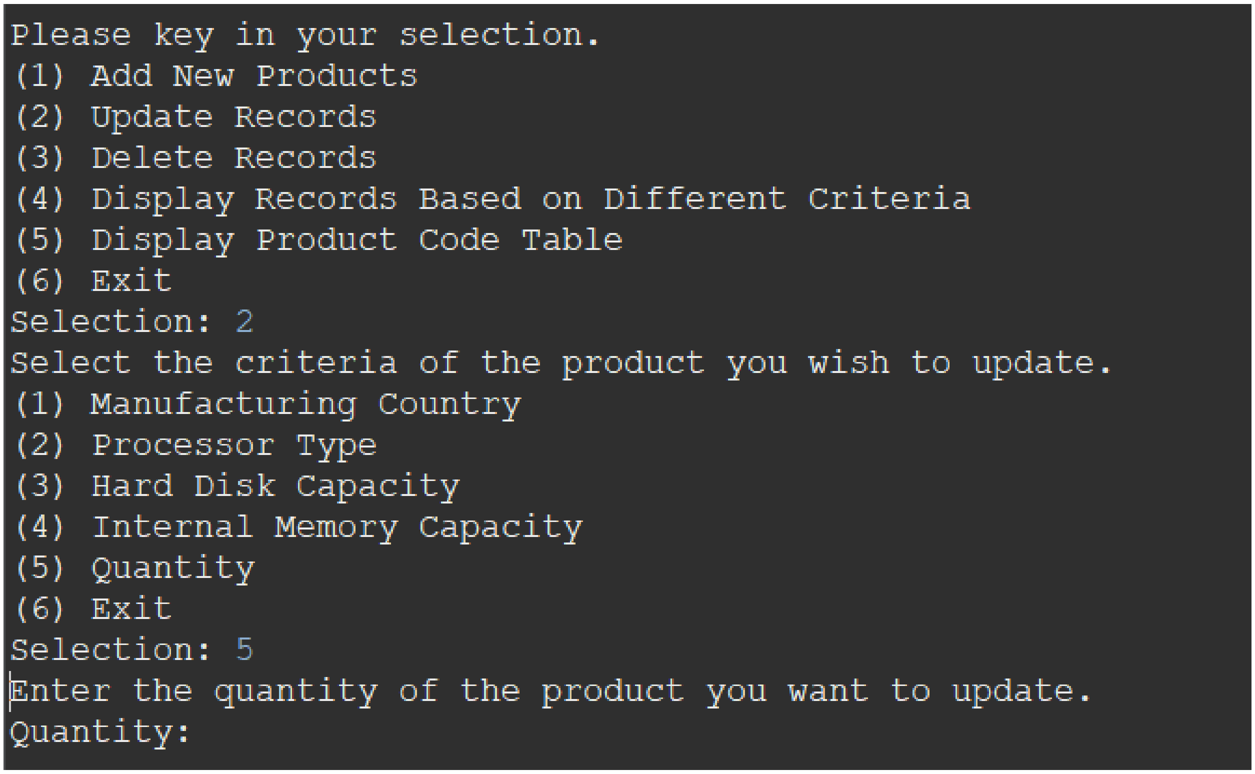
```
Select the internal memory capacity of the product you want to update.
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection: 1
Select the new internal memory capacity:
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection:
```

Figure 28 - 2 Update (4) Internal Memory Capacity



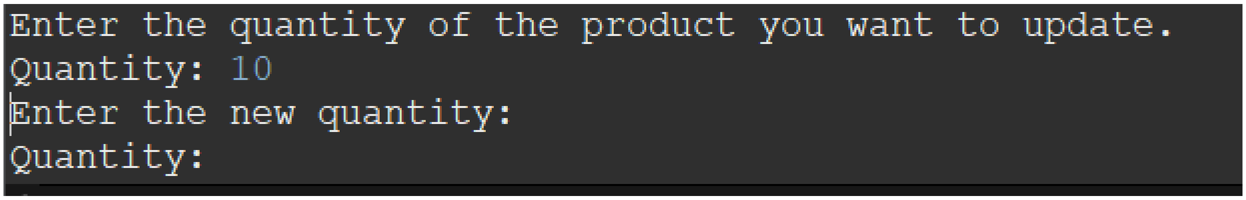
```
Select the new internal memory capacity:
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection: 2
There are 5 records with internal memory capacity of 1 TB:
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Malaysia         Intel i5           320 GB                      1 TB                           10
J53201024       Japan            Intel i5           320 GB                      1 TB                           5
A53201024       America          Intel i5           320 GB                      1 TB                           20
M73201024       Malaysia         Intel i7           320 GB                      1 TB                           10
A910241024      America          Intel i9           1 TB                        1 TB                           10
(1) Update All
(2) Update One
Selection:
```

Figure 28 - 3 Update (4) Internal Memory Capacity



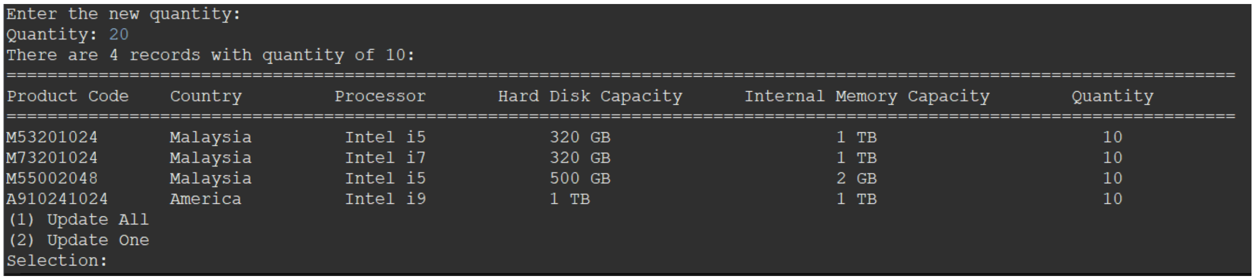
```
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 2
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection: 5
Enter the quantity of the product you want to update.
Quantity:
```

Figure 29 - 1 Update (5) Quantity



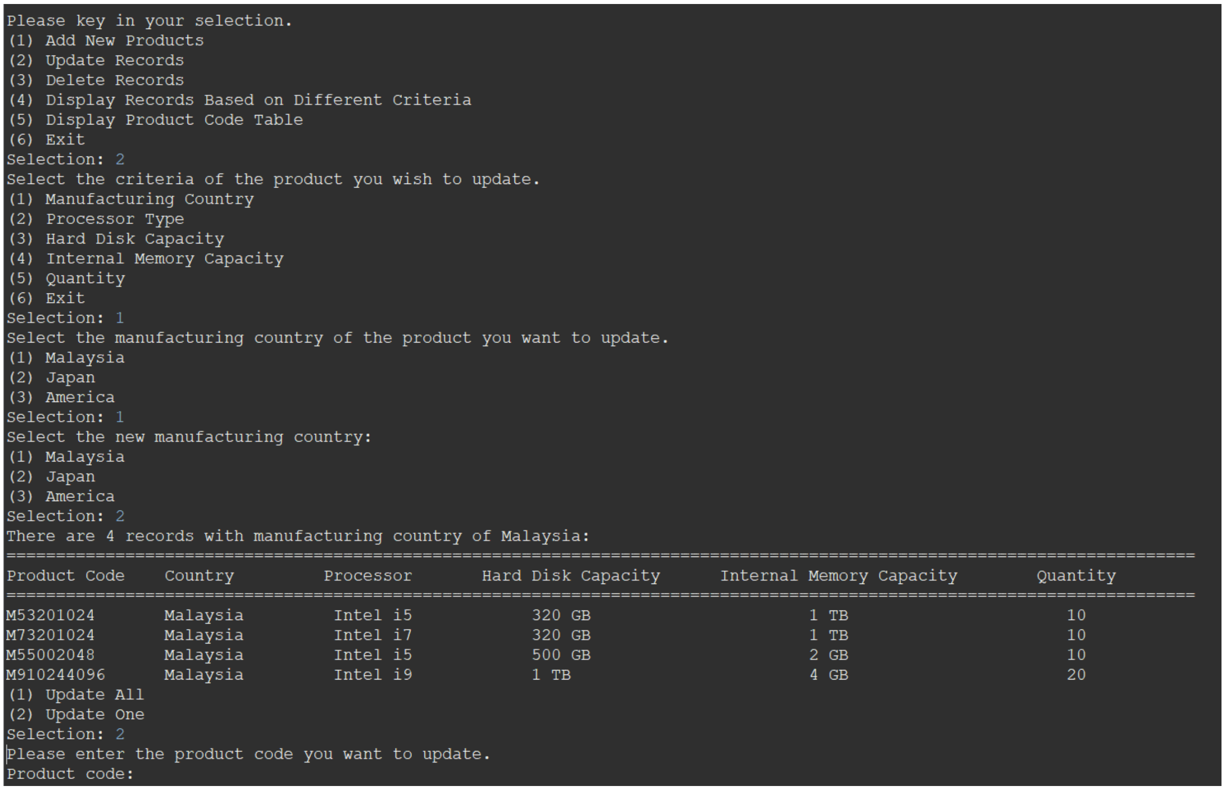
```
Enter the quantity of the product you want to update.
Quantity: 10
Enter the new quantity:
Quantity:
```

Figure 29 - 2 Update (5) Quantity



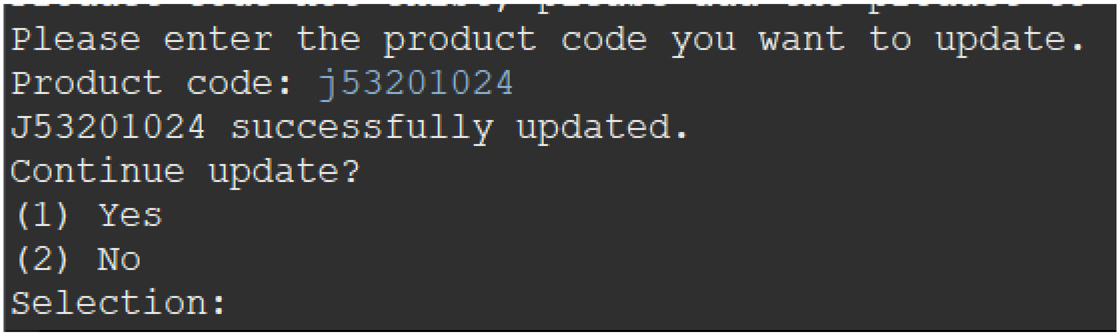
```
Enter the new quantity:
Quantity: 20
There are 4 records with quantity of 10:
==========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
==========================================================================================================================
M53201024       Malaysia         Intel i5          320 GB                          1 TB                       10
M73201024       Malaysia         Intel i7          320 GB                          1 TB                       10
M55002048       Malaysia         Intel i5          500 GB                          2 GB                       10
A910241024      America          Intel i9          1 TB                            1 TB                       10
(1) Update All
(2) Update One
Selection:
```

Figure 29 - 3 Update (5) Quantity



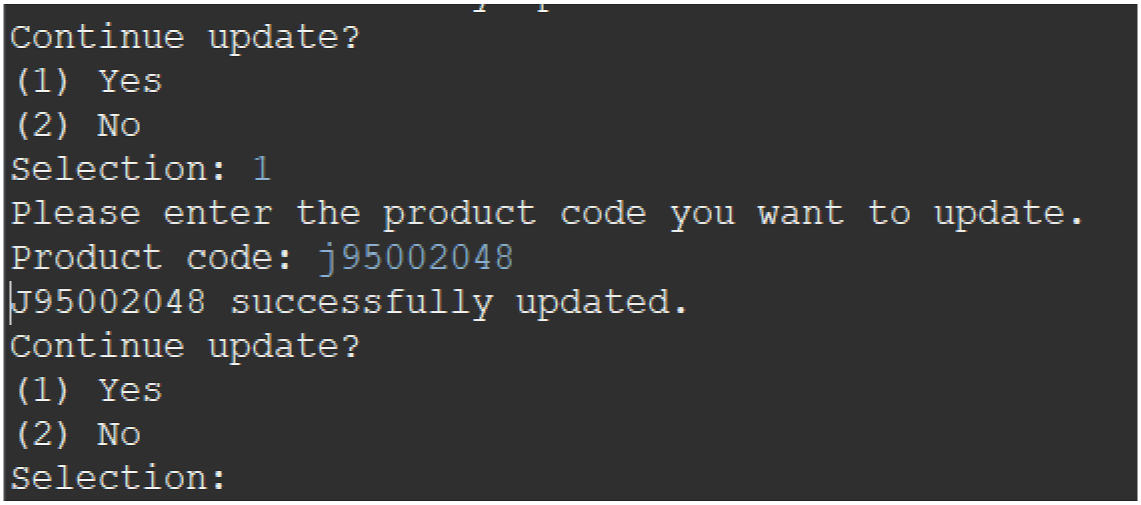
```
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 2
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection: 1
Select the manufacturing country of the product you want to update.
(1) Malaysia
(2) Japan
(3) America
Selection: 1
Select the new manufacturing country:
(1) Malaysia
(2) Japan
(3) America
Selection: 2
There are 4 records with manufacturing country of Malaysia:
==========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
==========================================================================================================================
M53201024       Malaysia         Intel i5          320 GB                          1 TB                       10
M73201024       Malaysia         Intel i7          320 GB                          1 TB                       10
M55002048       Malaysia         Intel i5          500 GB                          2 GB                       10
M910244096      Malaysia         Intel i9          1 TB                            4 GB                       20
(1) Update All
(2) Update One
Selection: 2
Please enter the product code you want to update.
Product code:
```

Figure 30 - 1 (2) Update One



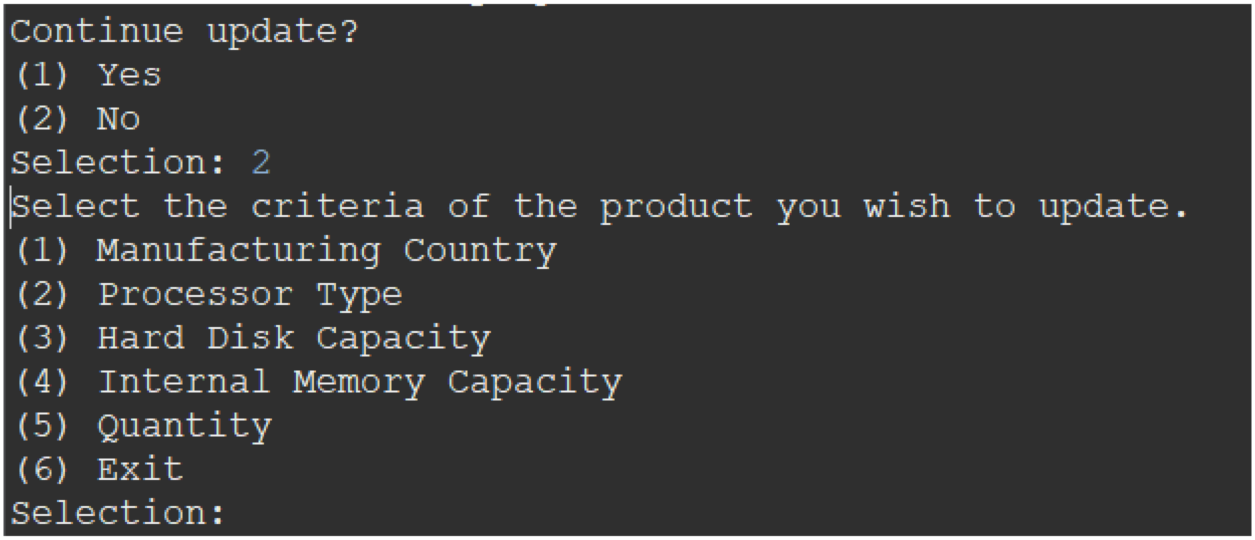
```
Please enter the product code you want to update.
Product code: j53201024
J53201024 successfully updated.
Continue update?
(1) Yes
(2) No
Selection:
```

Figure 30 - 2 (2) Update One



```
Continue update?
(1) Yes
(2) No
Selection: 1
Please enter the product code you want to update.
Product code: j95002048
J95002048 successfully updated.
Continue update?
(1) Yes
(2) No
Selection:
```

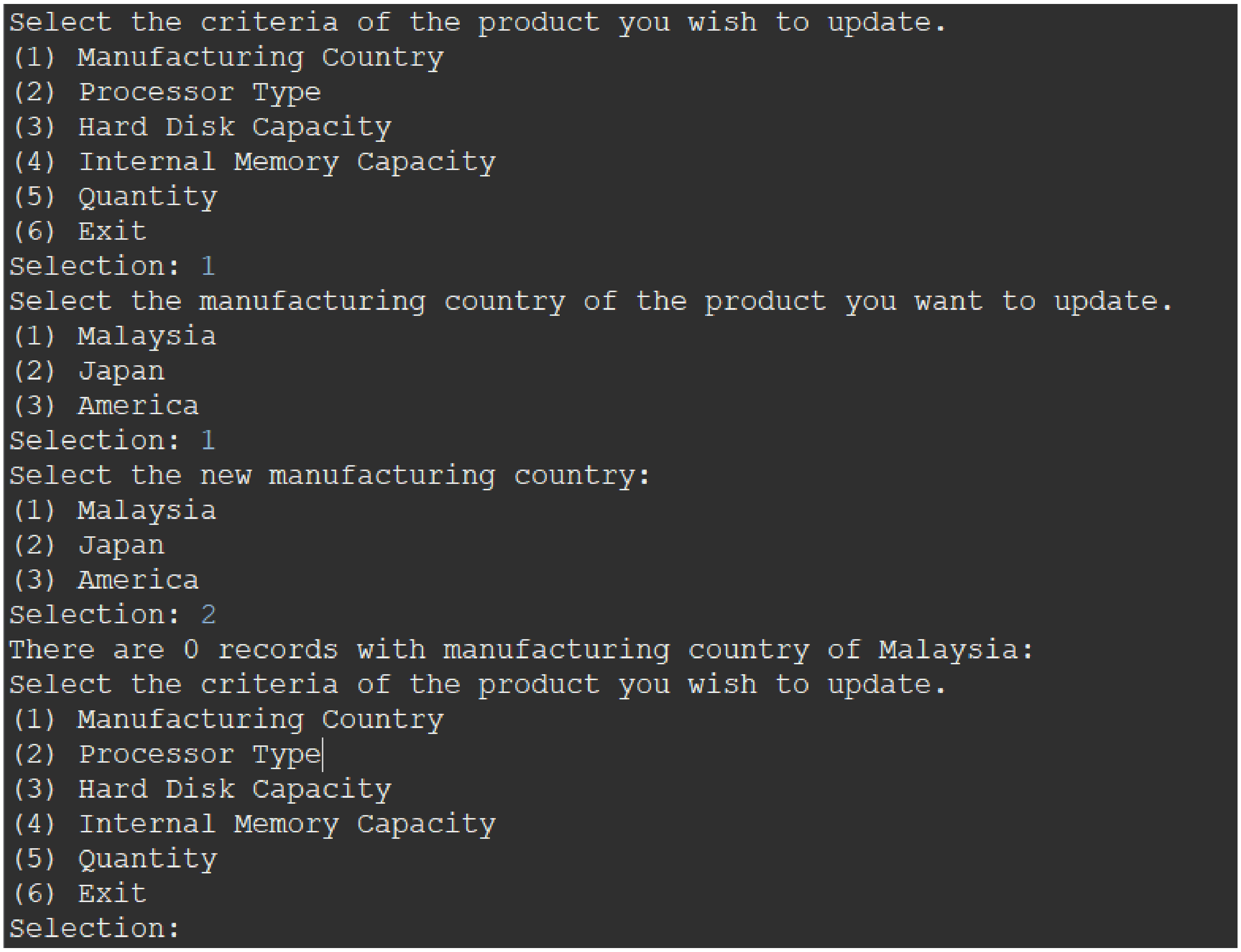
Figure 30 - 3 (2) Update One – Continue Update



```
Continue update?
(1) Yes
(2) No
Selection: 2
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection:
```

Figure 30 – 4 (2) Update One – No Continue Update

If user select a criteria that no records match the criteria for updation, the system will prompt that, there are 0 records for that particular criteria and then direct the user back to the update menu as shown in *Figure 31*.

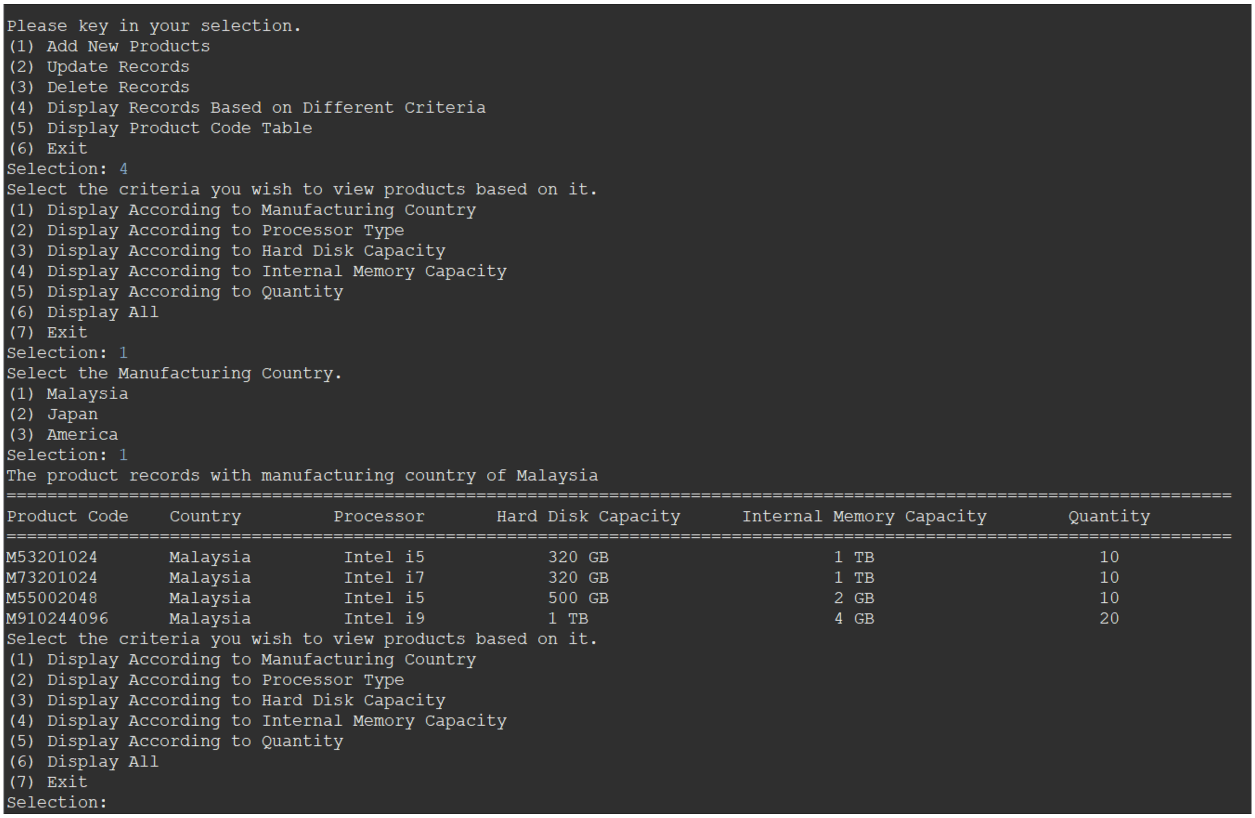


```
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection: 1
Select the manufacturing country of the product you want to update.
(1) Malaysia
(2) Japan
(3) America
Selection: 1
Select the new manufacturing country:
(1) Malaysia
(2) Japan
(3) America
Selection: 2
There are 0 records with manufacturing country of Malaysia:
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection:
```

Figure 31 No record with the chose criteria to update

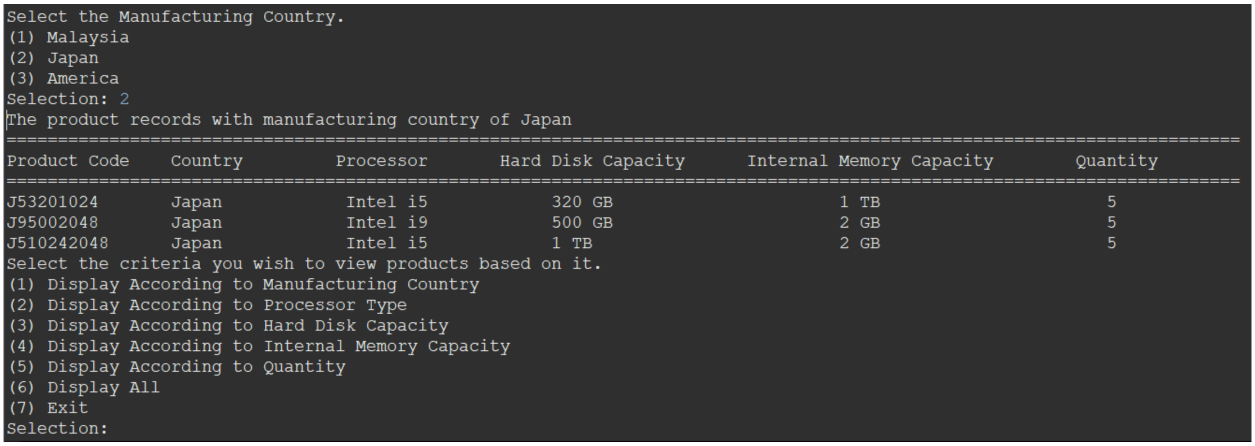
(4) Display Records Based on Different Criteria

(1) Display According to Manufacturing Country



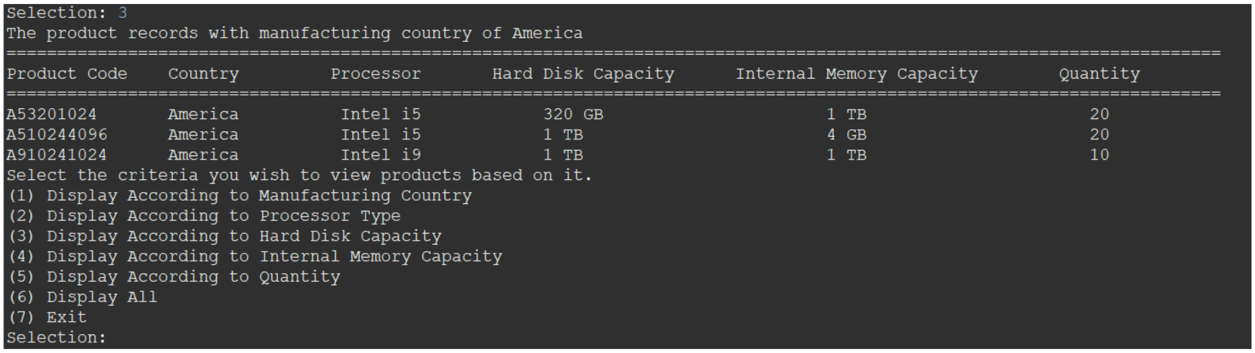
```
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 4
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection: 1
Select the Manufacturing Country.
(1) Malaysia
(2) Japan
(3) America
Selection: 1
The product records with manufacturing country of Malaysia
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Malaysia         Intel i5            320 GB                      1 TB                      10
M73201024       Malaysia         Intel i7            320 GB                      1 TB                      10
M55002048       Malaysia         Intel i5            500 GB                      2 GB                      10
M910244096      Malaysia         Intel i9            1 TB                        4 GB                      20
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

Figure 32 – 1 (1) Malaysia



```
Select the Manufacturing Country.
(1) Malaysia
(2) Japan
(3) America
Selection: 2
The product records with manufacturing country of Japan
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
J53201024       Japan            Intel i5            320 GB                      1 TB                      5
J95002048       Japan            Intel i9            500 GB                      2 GB                      5
J510242048      Japan            Intel i5            1 TB                        2 GB                      5
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

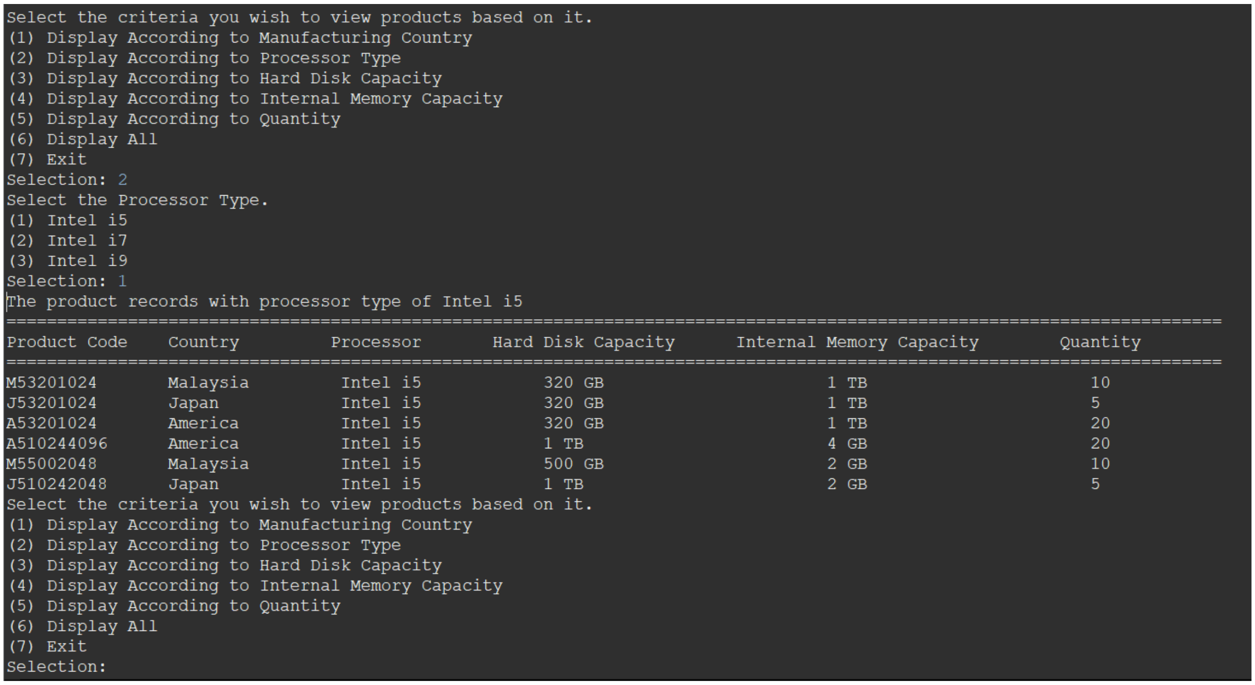
Figure 32 – 2 (2) Japan



```
Selection: 3
The product records with manufacturing country of America
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
A53201024       America          Intel i5            320 GB                      1 TB                      20
A510244096      America          Intel i5            1 TB                        4 GB                      20
A910241024      America          Intel i9            1 TB                        1 TB                      10
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

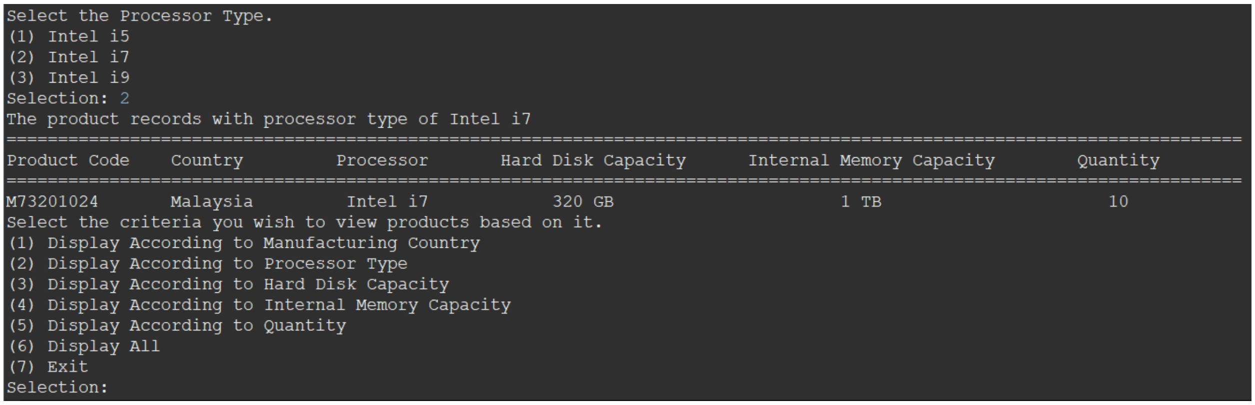
Figure 32 – 3 (3) America

<u>(2) Display According to Processor Type</u>



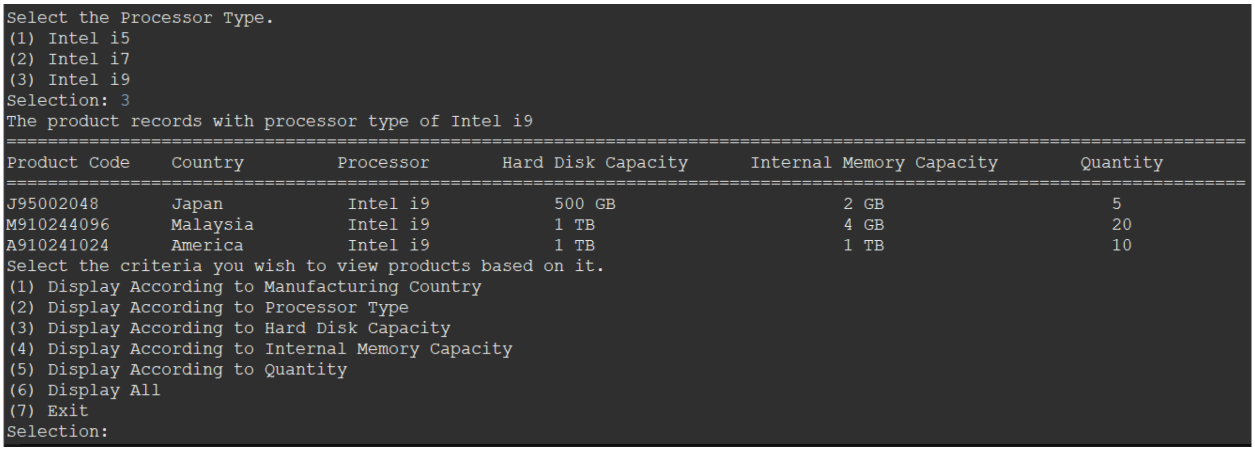
```
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection: 2
Select the Processor Type.
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection: 1
The product records with processor type of Intel i5
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Malaysia         Intel i5            320 GB                      1 TB                      10
J53201024       Japan            Intel i5            320 GB                      1 TB                      5
A53201024       America          Intel i5            320 GB                      1 TB                      20
A510244096      America          Intel i5            1 TB                        4 GB                      20
M55002048       Malaysia         Intel i5            500 GB                      2 GB                      10
J510242048      Japan            Intel i5            1 TB                        2 GB                      5
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

Figure 33 – 1 (1) Intel i5



```
Select the Processor Type.
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection: 2
The product records with processor type of Intel i7
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M73201024       Malaysia         Intel i7            320 GB                      1 TB                      10
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

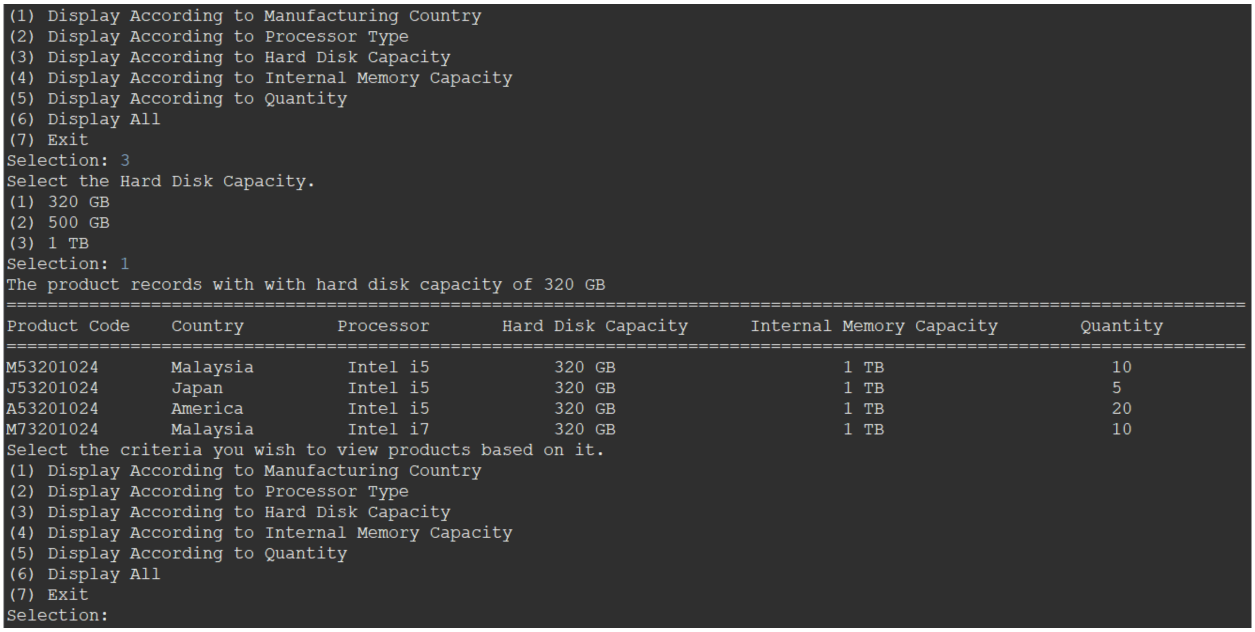
Figure 33 – 2 (1) Intel i7



```
Select the Processor Type.
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection: 3
The product records with processor type of Intel i9
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
J95002048       Japan            Intel i9            500 GB                      2 GB                      5
M910244096      Malaysia         Intel i9            1 TB                        4 GB                      20
A910241024      America          Intel i9            1 TB                        1 TB                      10
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

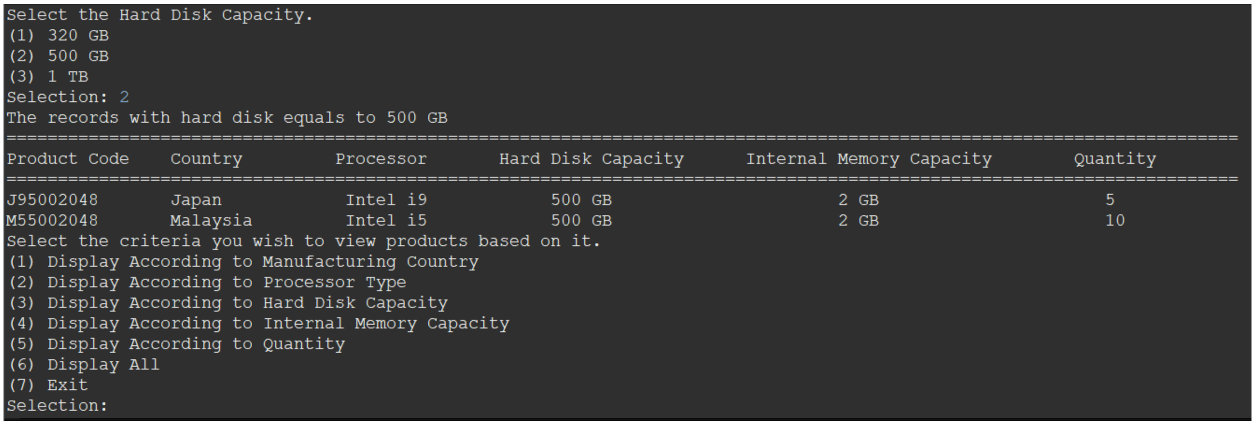
Figure 33 – 3 (1) Intel i9

(3) Display According to Hard Disk Capacity



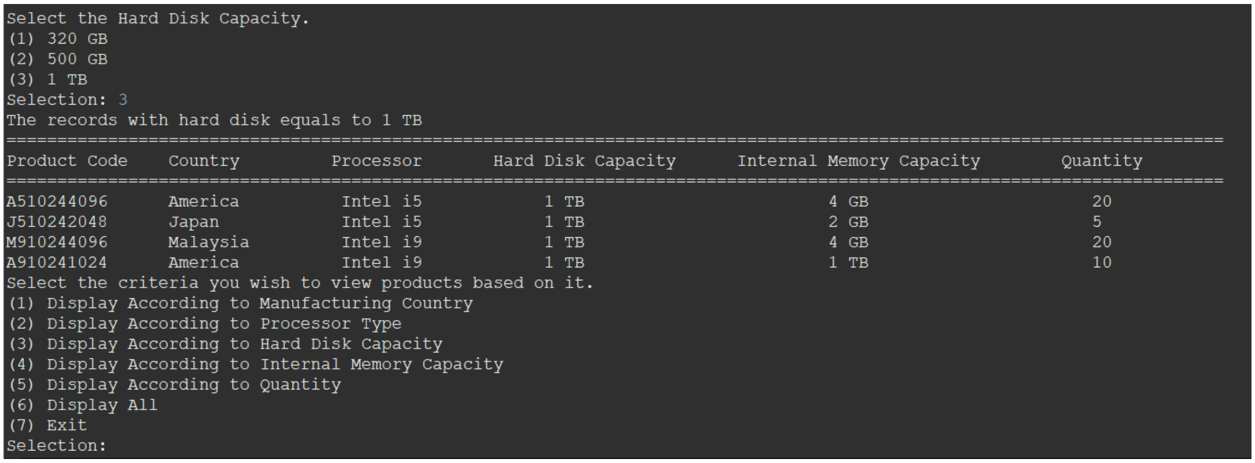
```
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection: 3
Select the Hard Disk Capacity.
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection: 1
The product records with with hard disk capacity of 320 GB
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Malaysia         Intel i5            320 GB                      1 TB                      10
J53201024       Japan            Intel i5            320 GB                      1 TB                      5
A53201024       America          Intel i5            320 GB                      1 TB                      20
M73201024       Malaysia         Intel i7            320 GB                      1 TB                      10
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

Figure 34 – 1 (1) 320 GB



```
Select the Hard Disk Capacity.
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection: 2
The records with hard disk equals to 500 GB
=========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity         Quantity
=========================================================================================================================
J95002048       Japan            Intel i9           500 GB                      2 GB                       5
M55002048       Malaysia         Intel i5           500 GB                      2 GB                       10
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

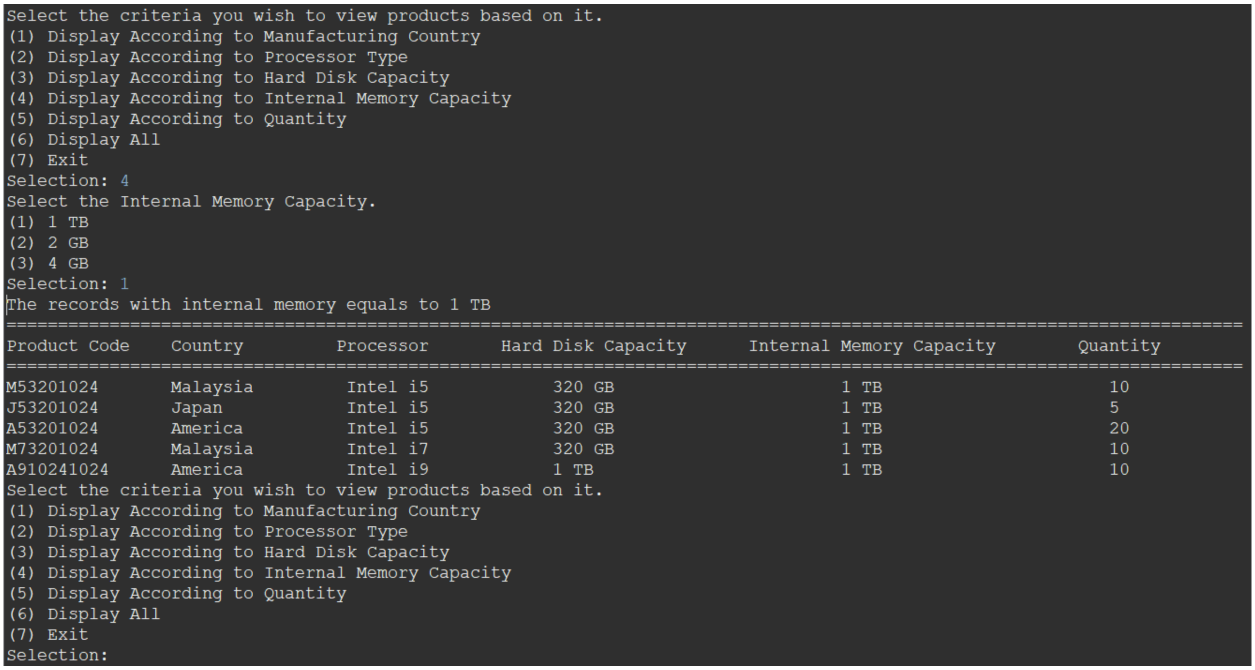
Figure 34 – 2 (2) 500 GB



```
Select the Hard Disk Capacity.
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection: 3
The records with hard disk equals to 1 TB
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity       Quantity
========================================================================================================================
A510244096      America          Intel i5           1 TB                        4 GB                     20
J510242048      Japan            Intel i5           1 TB                        2 GB                     5
M910244096      Malaysia         Intel i9           1 TB                        4 GB                     20
A910241024      America          Intel i9           1 TB                        1 TB                     10
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

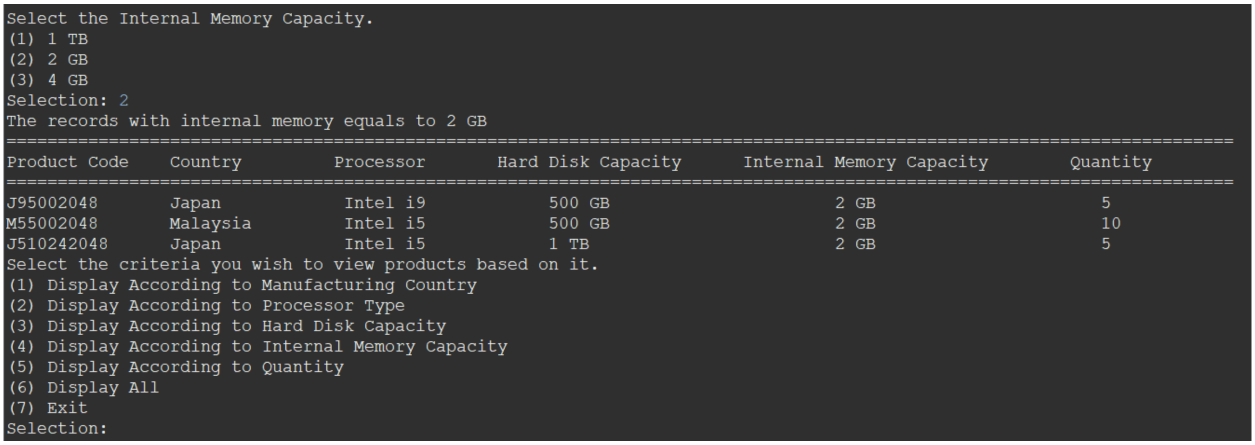
Figure 34 – 3 (3) 1 TB

(4) Display According to Internal Memory Capacity



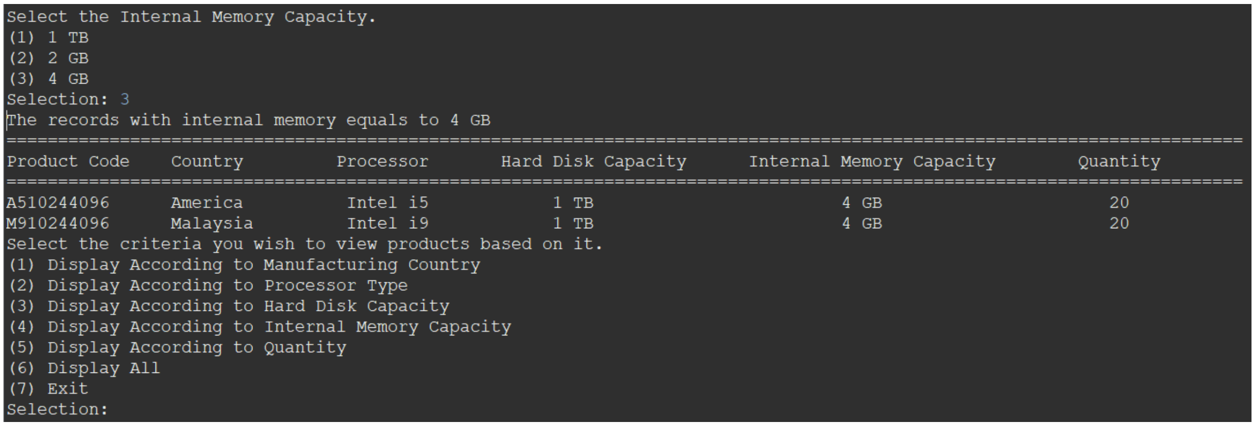
```
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection: 4
Select the Internal Memory Capacity.
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection: 1
The records with internal memory equals to 1 TB
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Malaysia         Intel i5           320 GB                      1 TB                          10
J53201024       Japan            Intel i5           320 GB                      1 TB                          5
A53201024       America          Intel i5           320 GB                      1 TB                          20
M73201024       Malaysia         Intel i7           320 GB                      1 TB                          10
A910241024      America          Intel i9           1 TB                        1 TB                          10
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

Figure 35 – 1 (1) 1 TB



```
Select the Internal Memory Capacity.
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection: 2
The records with internal memory equals to 2 GB
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
J95002048       Japan            Intel i9           500 GB                      2 GB                           5
M55002048       Malaysia         Intel i5           500 GB                      2 GB                           10
J510242048      Japan            Intel i5           1 TB                        2 GB                           5
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

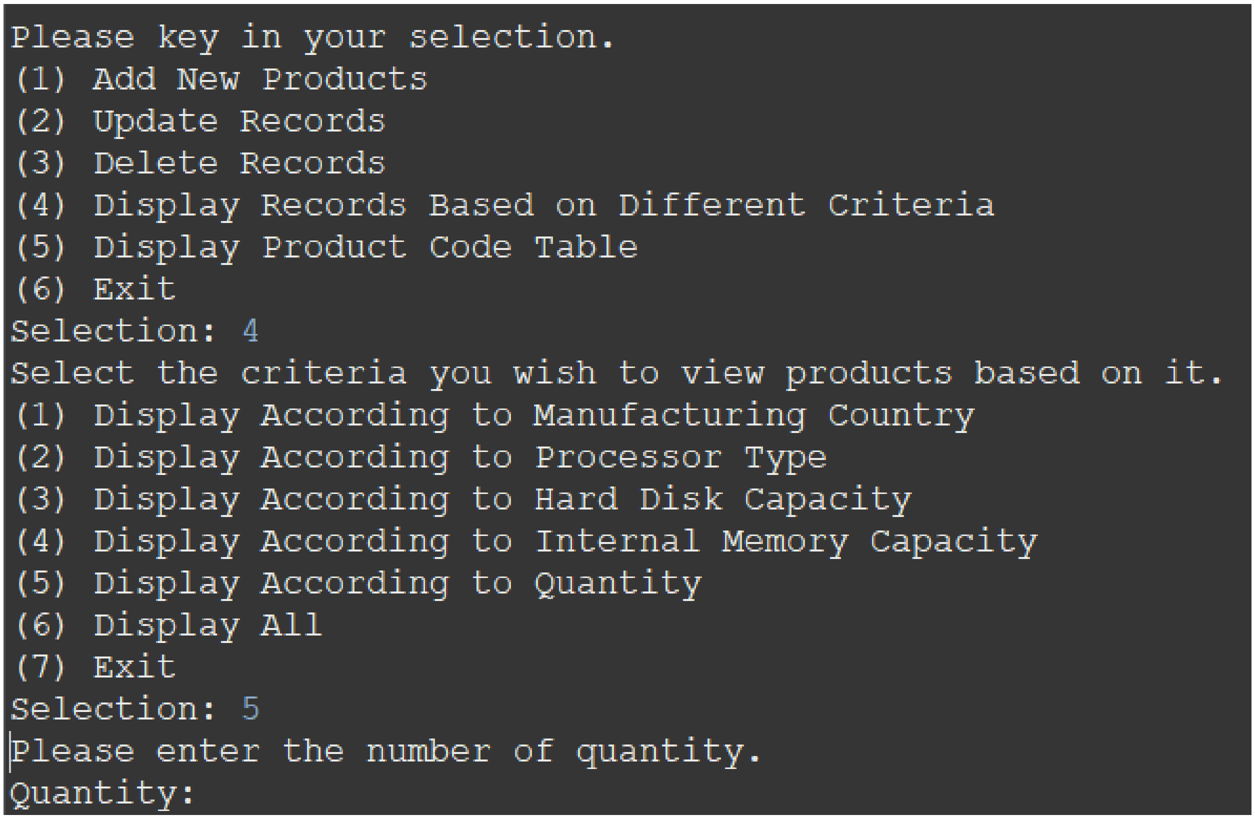
Figure 35 – 2 (2) 2 GB



```
Select the Internal Memory Capacity.
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection: 3
The records with internal memory equals to 4 GB
===============================================================================================================================
Product Code     Country          Processor        Hard Disk Capacity       Internal Memory Capacity          Quantity
===============================================================================================================================
A510244096       America           Intel i5             1 TB                         4 GB                        20
M910244096       Malaysia          Intel i9             1 TB                         4 GB                        20
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

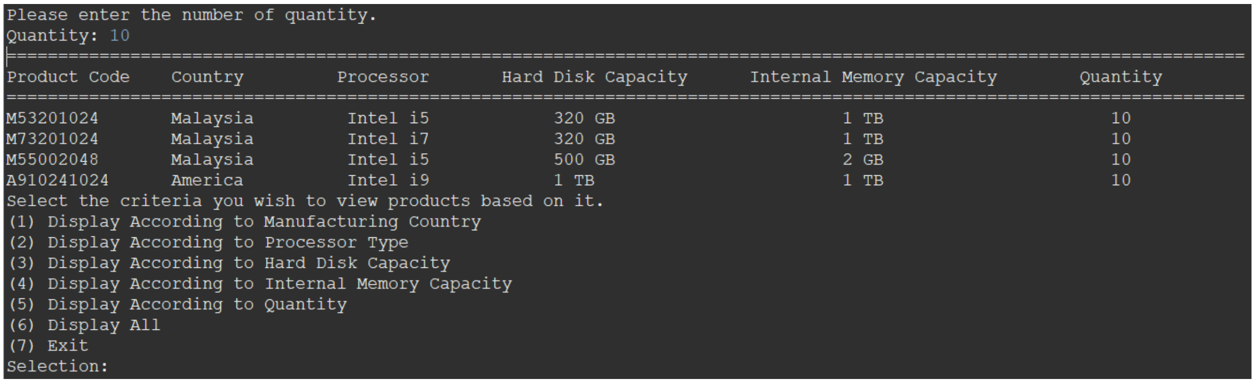
Figure 35 – 3 (3) 4 GB

(5) Display According to Quantity



```
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 4
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection: 5
Please enter the number of quantity.
Quantity:
```

Figure 36 – 1 Display According to Quantity



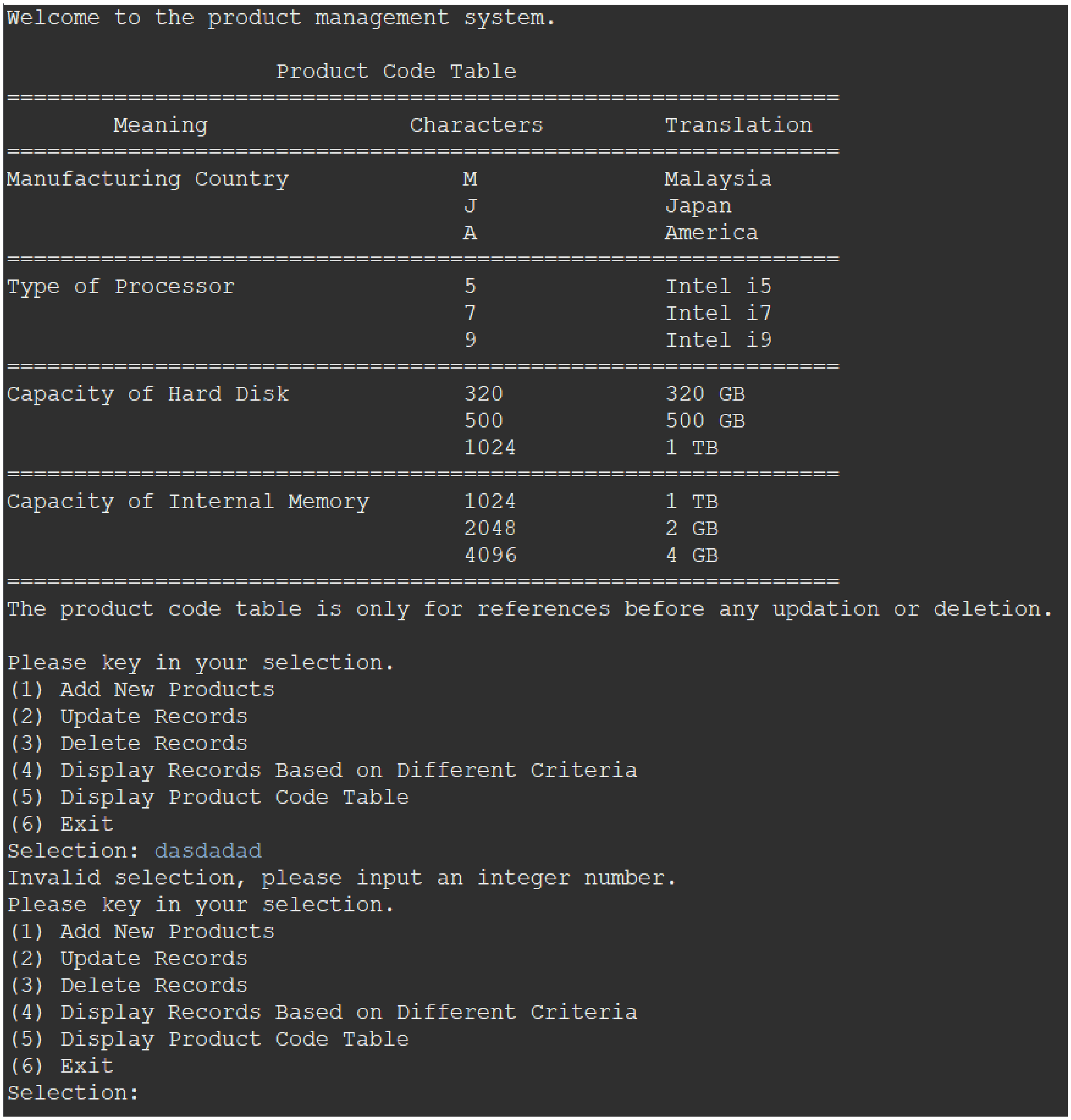
```
Please enter the number of quantity.
Quantity: 10
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Malaysia         Intel i5           320 GB                       1 TB                      10
M73201024       Malaysia         Intel i7           320 GB                       1 TB                      10
M55002048       Malaysia         Intel i5           500 GB                       2 GB                      10
A910241024      America          Intel i9           1 TB                         1 TB                      10
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

Figure 36 – 2 Display According to Quantity

1.2.3 Validation of the Program

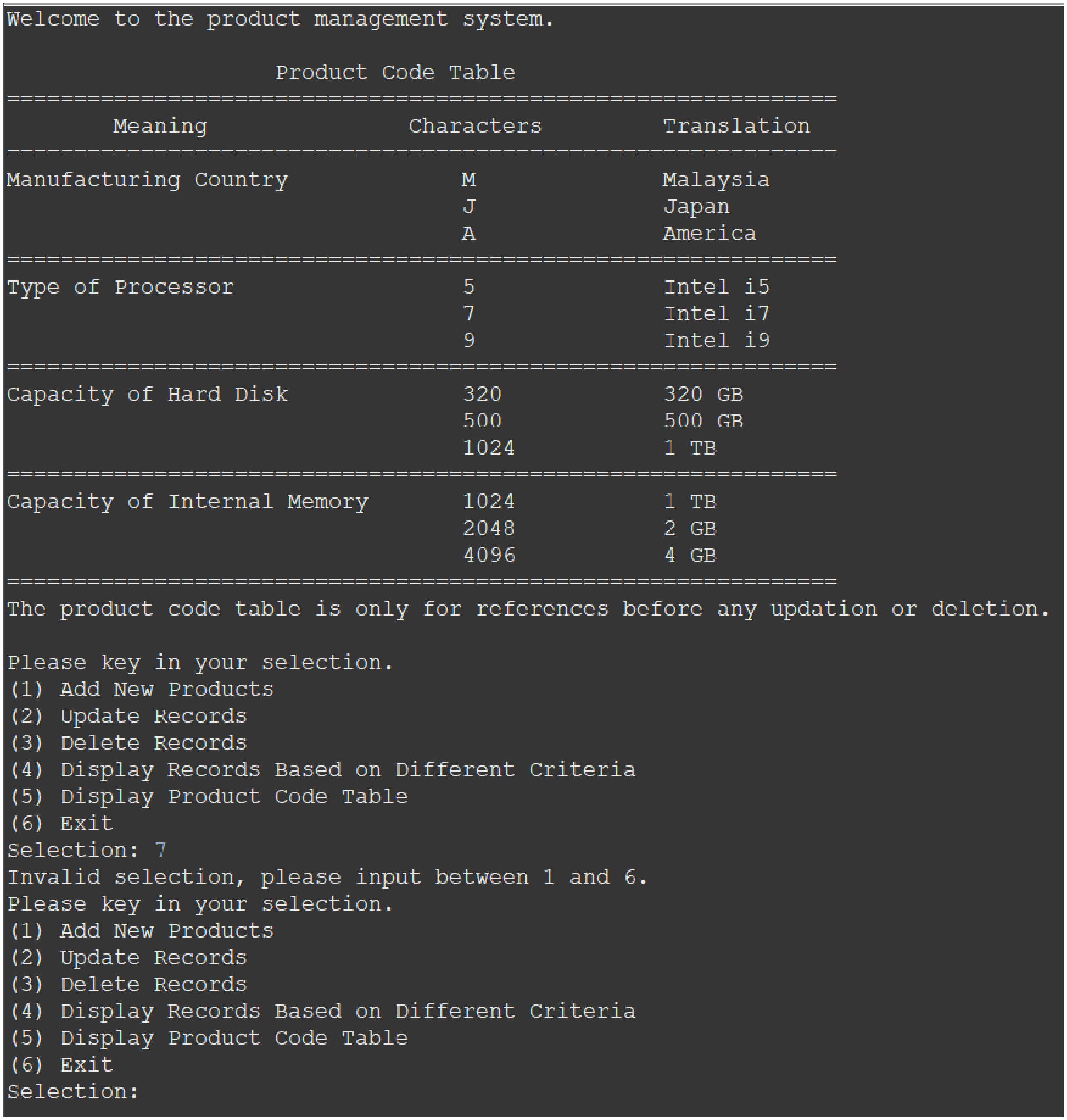
The validation is used to show the validation system provided with system.

1. **Main Menu Validation**
   If user enter an invalid selection which is contain non-numeric character or out of selection range, the system will prompt out the error message and ask the user to enter the selection again as shown in *Figure 37*, *Figure 38*.



```
Welcome to the product management system.

                    Product Code Table
===============================================================
        Meaning                   Characters        Translation
===============================================================
Manufacturing Country              M                Malaysia
                                   J                Japan
                                   A                America
===============================================================
Type of Processor                  5                Intel i5
                                   7                Intel i7
                                   9                Intel i9
===============================================================
Capacity of Hard Disk              320              320 GB
                                   500              500 GB
                                   1024             1 TB
===============================================================
Capacity of Internal Memory        1024             1 TB
                                   2048             2 GB
                                   4096             4 GB
===============================================================
The product code table is only for references before any updation or deletion.

Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: dasdadad
Invalid selection, please input an integer number.
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection:
```

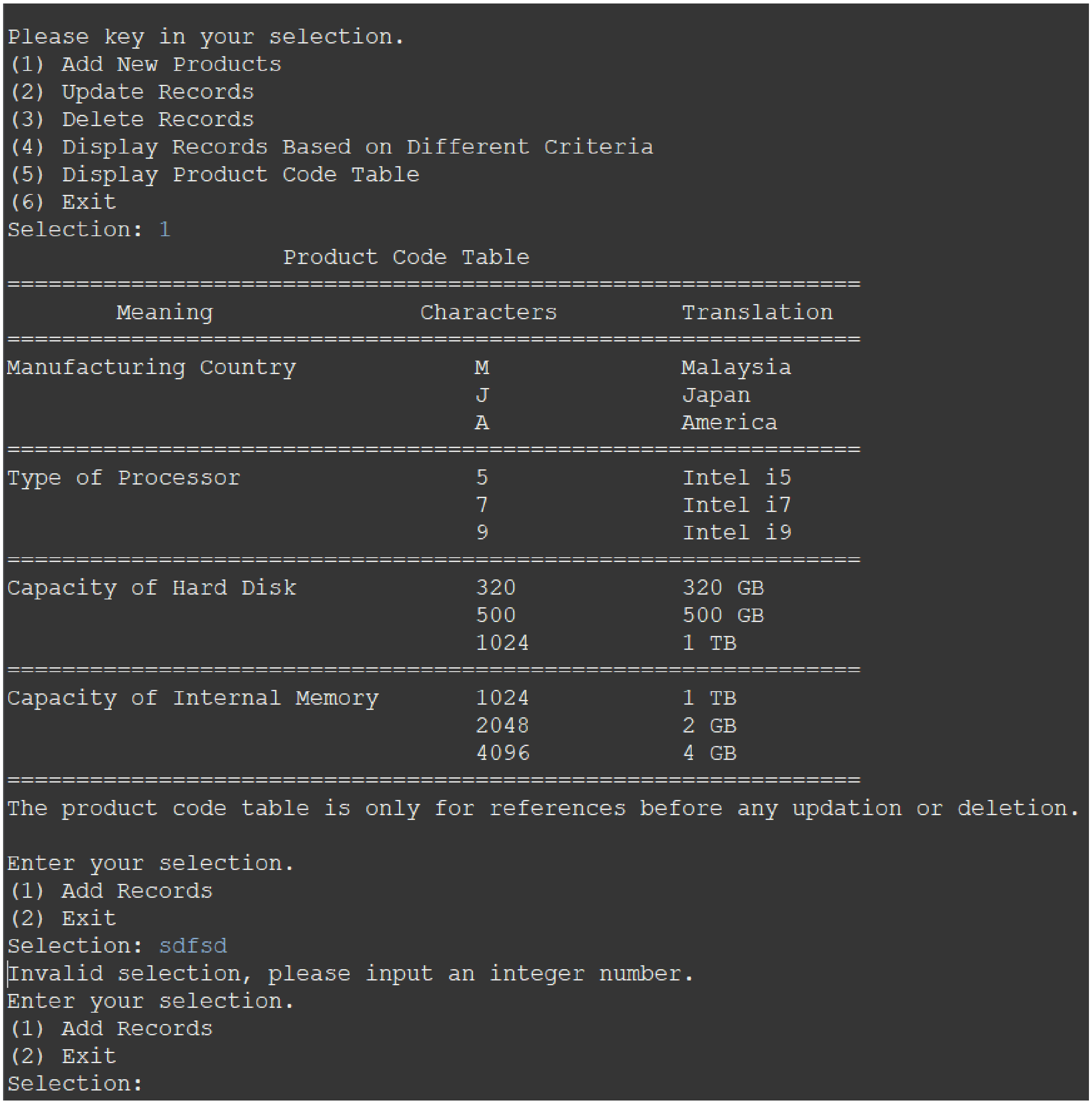
Figure 37 Invalid Selection – Non-Numeric Character



```
Welcome to the product management system.

                    Product Code Table
==============================================================
        Meaning               Characters         Translation
==============================================================
Manufacturing Country             M              Malaysia
                                  J              Japan
                                  A              America
==============================================================
Type of Processor                 5              Intel i5
                                  7              Intel i7
                                  9              Intel i9
==============================================================
Capacity of Hard Disk             320            320 GB
                                  500            500 GB
                                  1024           1 TB
==============================================================
Capacity of Internal Memory       1024           1 TB
                                  2048           2 GB
                                  4096           4 GB
==============================================================
The product code table is only for references before any updation or deletion.

Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 7
Invalid selection, please input between 1 and 6.
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection:
```

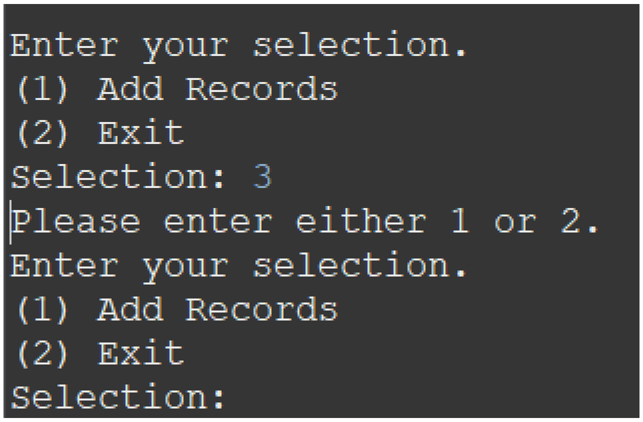
Figure 38 Invalid Selection – Not Within Selection Range

2. **(1) Add New Products Validation**
   If user enter an invalid selection which is contain non-numeric character or out of selection range, the system will prompt out the error message and ask the user to enter the selection again as shown in *Figure 39*, *Figure 40*.



```
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 1
                    Product Code Table
==============================================================
        Meaning               Characters        Translation
==============================================================
Manufacturing Country             M             Malaysia
                                  J             Japan
                                  A             America
==============================================================
Type of Processor                 5             Intel i5
                                  7             Intel i7
                                  9             Intel i9
==============================================================
Capacity of Hard Disk             320           320 GB
                                  500           500 GB
                                  1024          1 TB
==============================================================
Capacity of Internal Memory       1024          1 TB
                                  2048          2 GB
                                  4096          4 GB
==============================================================
The product code table is only for references before any updation or deletion.

Enter your selection.
(1) Add Records
(2) Exit
Selection: sdfsd
Invalid selection, please input an integer number.
Enter your selection.
(1) Add Records
(2) Exit
Selection:
```

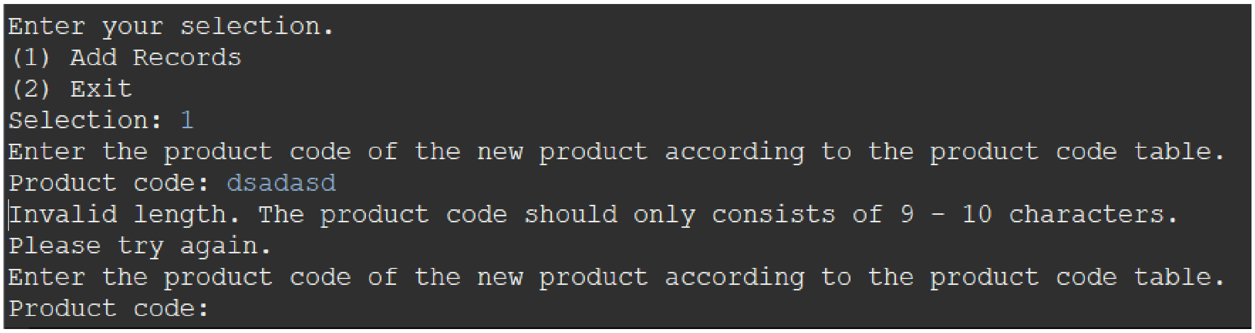
Figure 39 Invalid Selection – Non-Numeric Character



```
Enter your selection.
(1) Add Records
(2) Exit
Selection: 3
Please enter either 1 or 2.
Enter your selection.
(1) Add Records
(2) Exit
Selection:
```

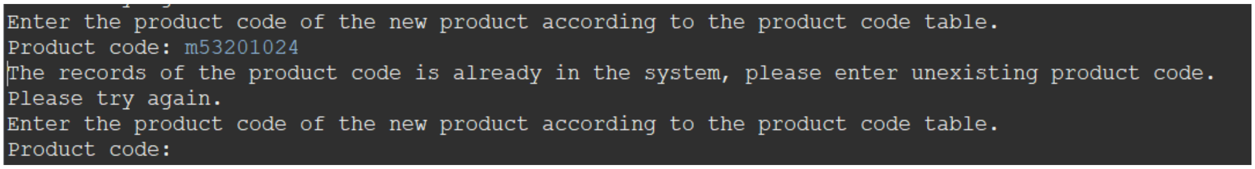
Figure 40 Invalid Selection – Not Within Selection Range

If user enter an invalid product code which is out of the range of the characters or does not exist in the system, the system will prompt out the error message and ask the user to enter the product code again as shown in *Figure 41*, *Figure 42*.



```
Enter your selection.
(1) Add Records
(2) Exit
Selection: 1
Enter the product code of the new product according to the product code table.
Product code: dsadasd
Invalid length. The product code should only consists of 9 - 10 characters.
Please try again.
Enter the product code of the new product according to the product code table.
Product code:
```

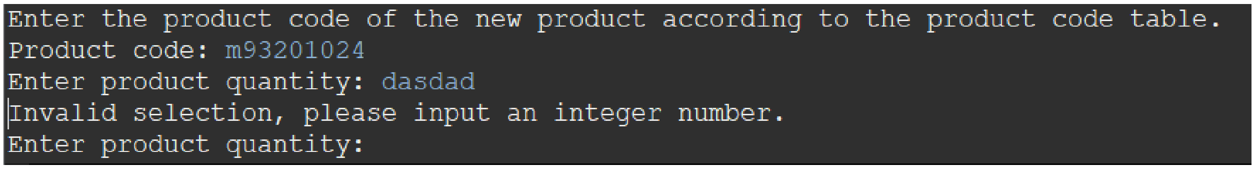
Figure 41 Invalid Product Code – Out of Range of Characters



```
Enter the product code of the new product according to the product code table.
Product code: m53201024
The records of the product code is already in the system, please enter unexisting product code.
Please try again.
Enter the product code of the new product according to the product code table.
Product code:
```

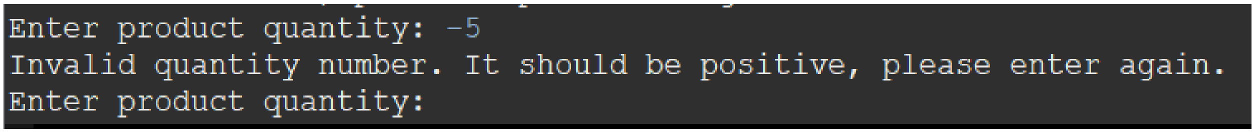
Figure 42 Invalid Product Code – Non-Existing Product Code

If user enter an invalid quantity which is contain non-numeric character or it is a negative value, the system will prompt out the error message and ask the user to enter the quantity again as shown in *Figure 43*, *Figure 44*.



```
Enter the product code of the new product according to the product code table.
Product code: m93201024
Enter product quantity: dasdad
Invalid selection, please input an integer number.
Enter product quantity:
```

Figure 43 Invalid Quantity – Non-Numeric Character

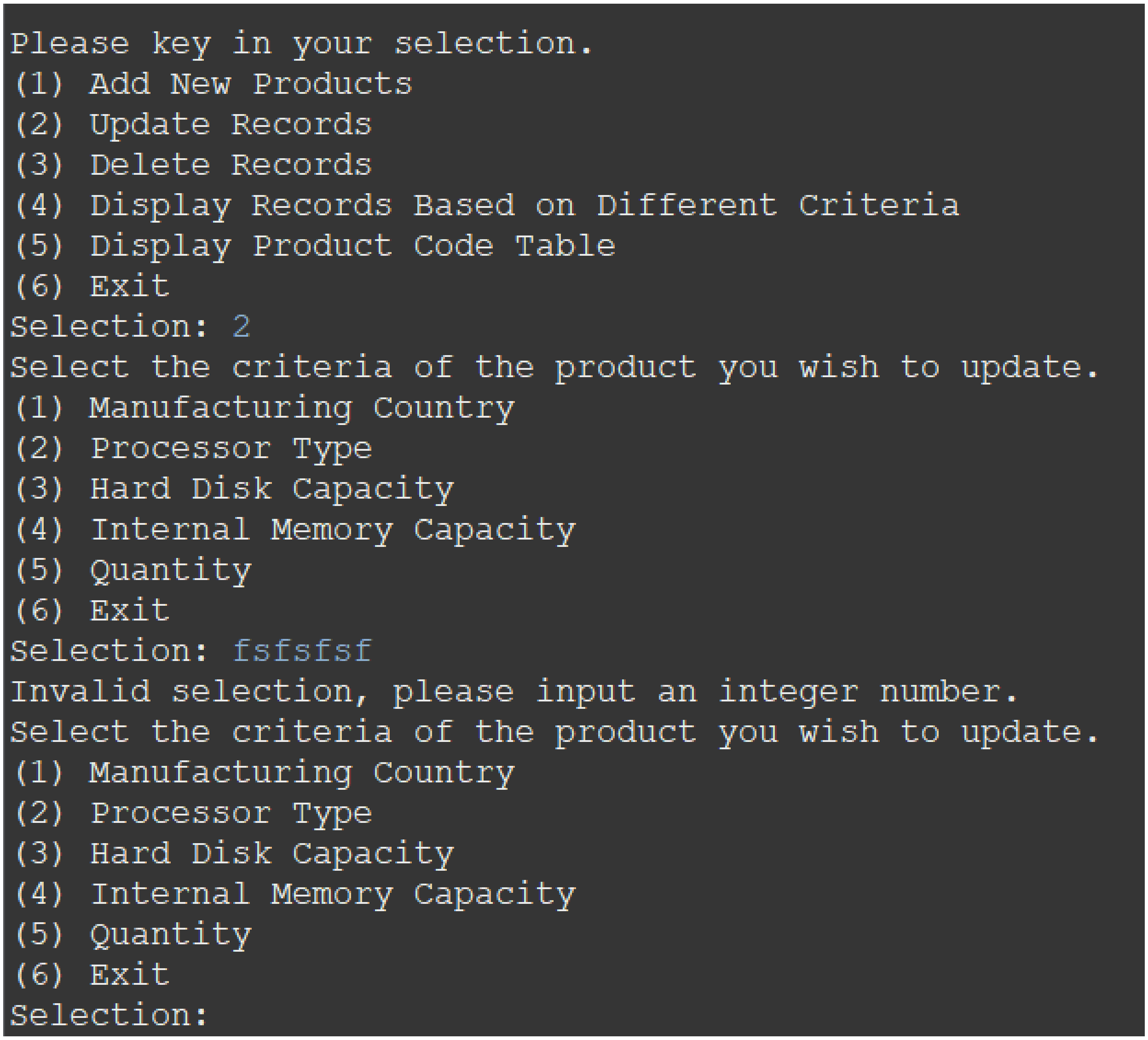


```
Enter product quantity: -5
Invalid quantity number. It should be positive, please enter again.
Enter product quantity:
```

Figure 44 Invalid Quantity – Negative Value

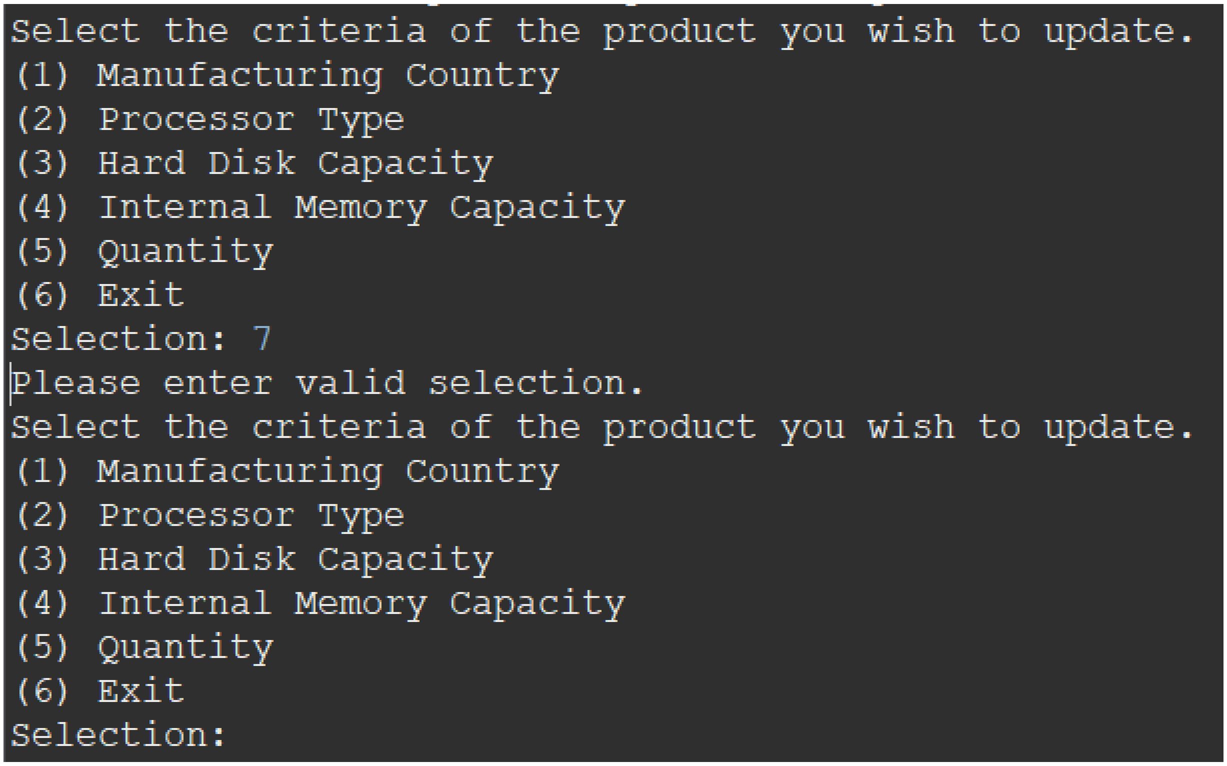
3. (2) Update Records Validation

If user enter an invalid selection which is contain non-numeric character or out of selection range, the system will prompt out the error message and ask the user to enter the selection again as shown in *Figure 45* to *Figure 65*.



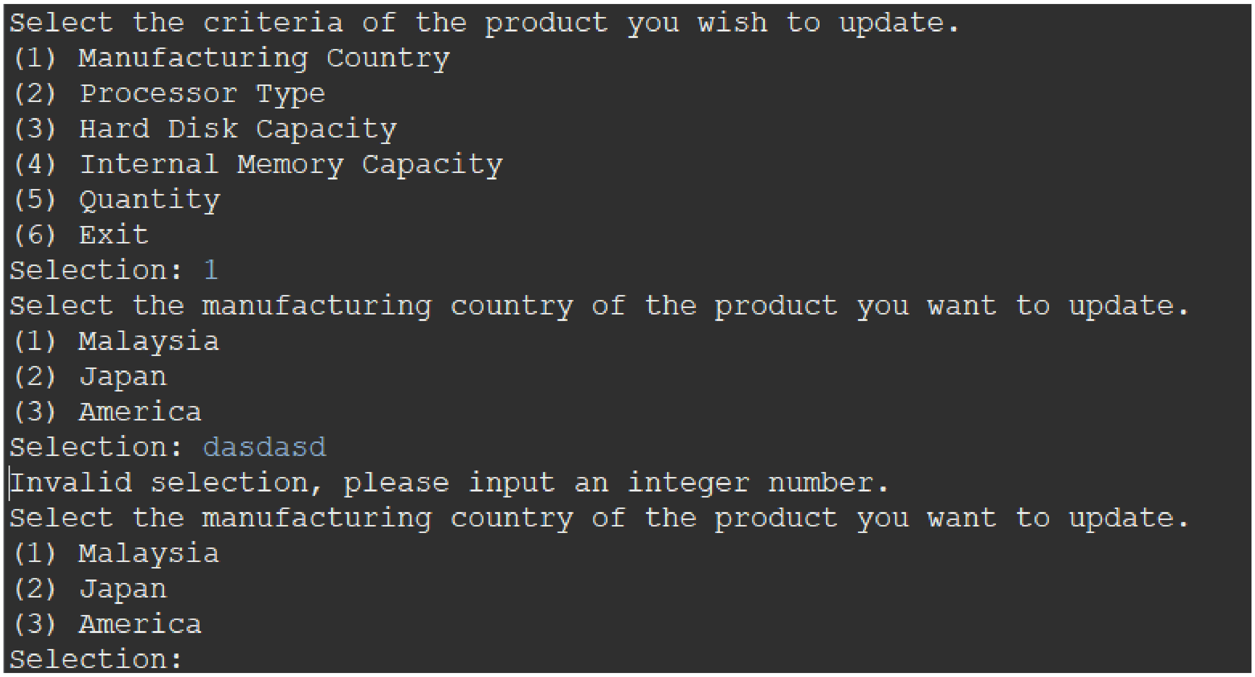
```
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 2
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection: fsfsfsf
Invalid selection, please input an integer number.
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection:
```

Figure 45 Invalid Selection – Non-Numeric Character



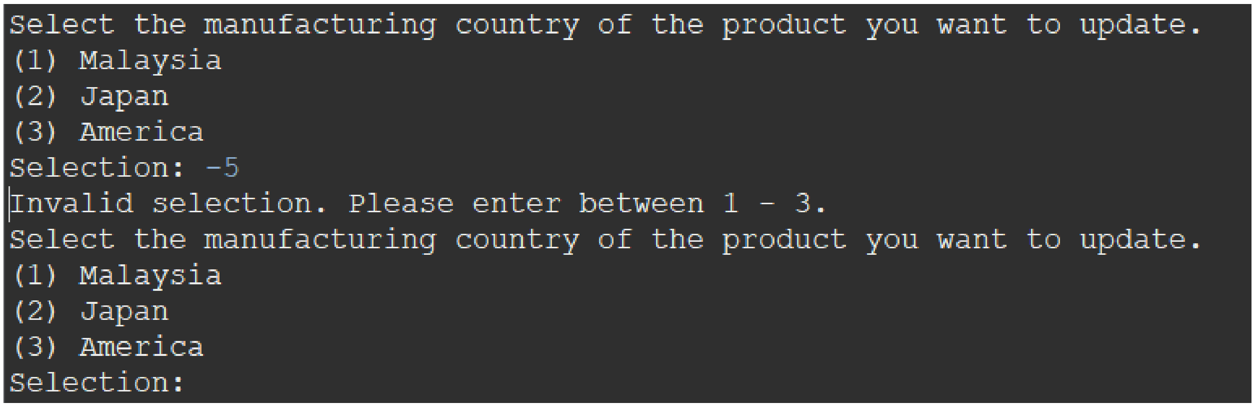
```
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection: 7
Please enter valid selection.
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection:
```

Figure 46 Invalid Selection – Not Within Selection Range



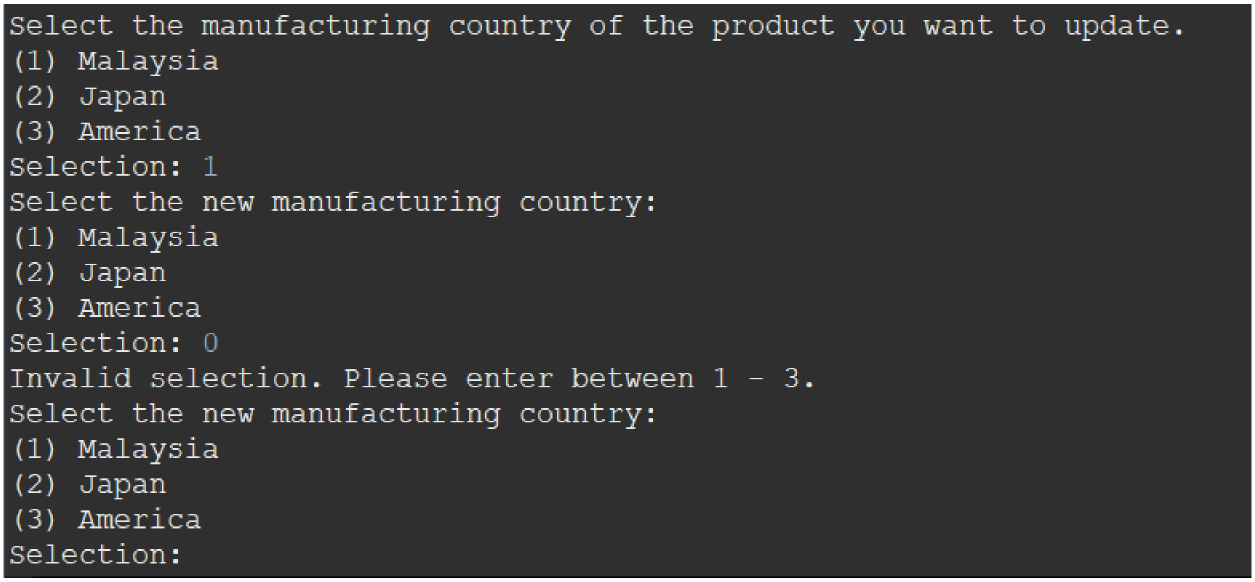
```
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection: 1
Select the manufacturing country of the product you want to update.
(1) Malaysia
(2) Japan
(3) America
Selection: dasdasd
Invalid selection, please input an integer number.
Select the manufacturing country of the product you want to update.
(1) Malaysia
(2) Japan
(3) America
Selection:
```

Figure 47 Invalid Selection – Non-Numeric Character



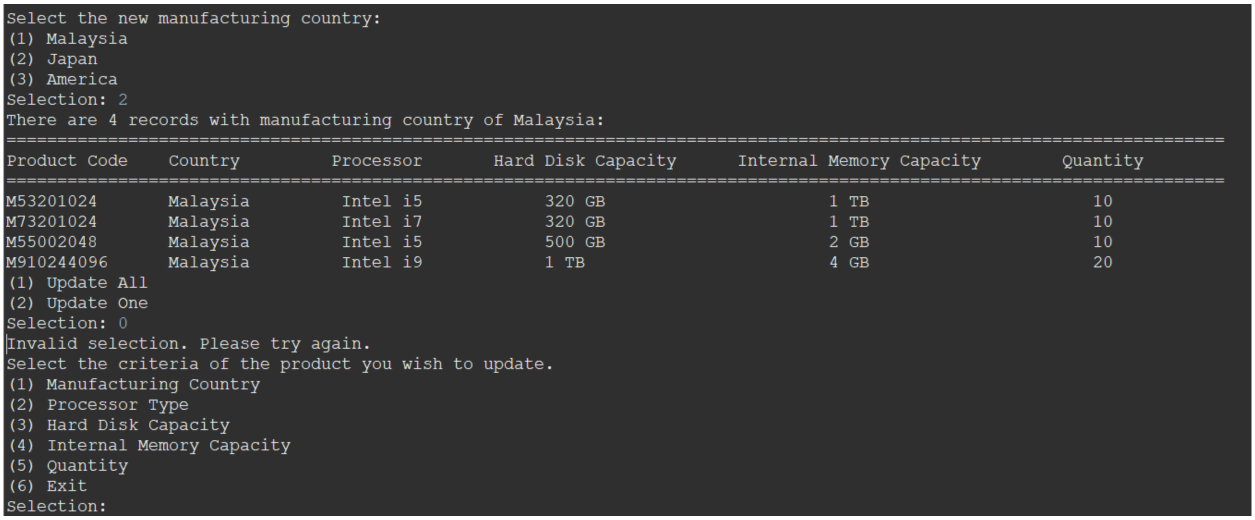
```
Select the manufacturing country of the product you want to update.
(1) Malaysia
(2) Japan
(3) America
Selection: -5
Invalid selection. Please enter between 1 - 3.
Select the manufacturing country of the product you want to update.
(1) Malaysia
(2) Japan
(3) America
Selection:
```

Figure 48 Invalid Selection – Not Within Selection Range



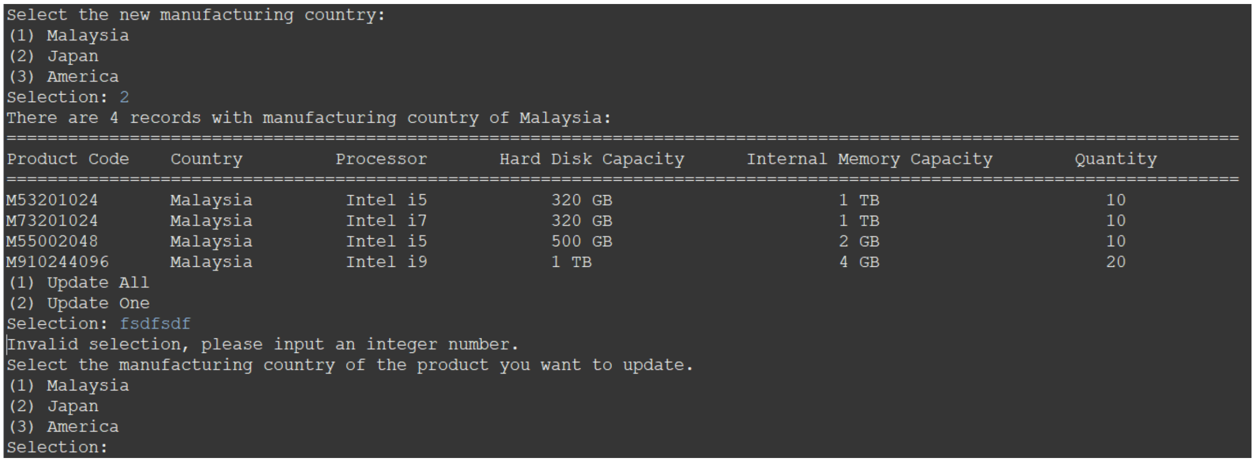
```
Select the manufacturing country of the product you want to update.
(1) Malaysia
(2) Japan
(3) America
Selection: 1
Select the new manufacturing country:
(1) Malaysia
(2) Japan
(3) America
Selection: 0
Invalid selection. Please enter between 1 - 3.
Select the new manufacturing country:
(1) Malaysia
(2) Japan
(3) America
Selection:
```

Figure 49 Invalid Selection – Not Within Selection Range



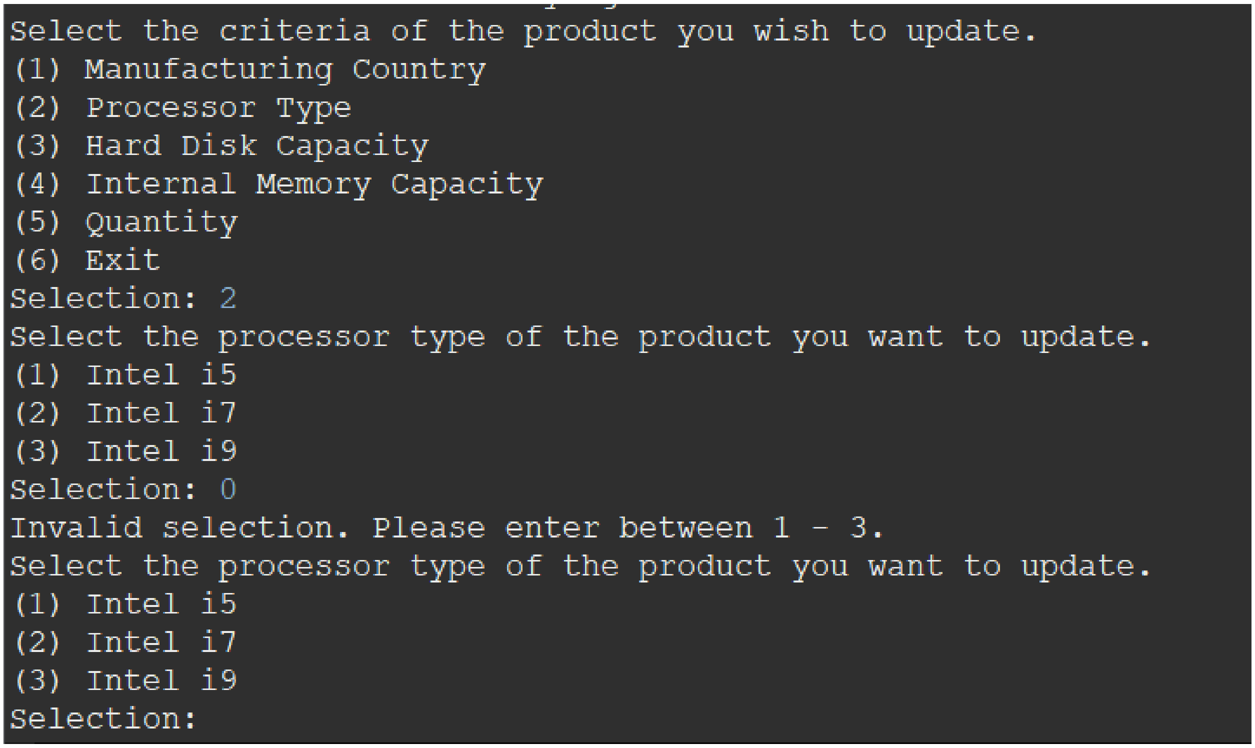
```
Select the new manufacturing country:
(1) Malaysia
(2) Japan
(3) America
Selection: 2
There are 4 records with manufacturing country of Malaysia:
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Malaysia         Intel i5           320 GB                      1 TB                       10
M73201024       Malaysia         Intel i7           320 GB                      1 TB                       10
M55002048       Malaysia         Intel i5           500 GB                      2 GB                       10
M910244096      Malaysia         Intel i9           1 TB                        4 GB                       20
(1) Update All
(2) Update One
Selection: 0
Invalid selection. Please try again.
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection:
```

Figure 50 Invalid Selection – Not Within Selection Range



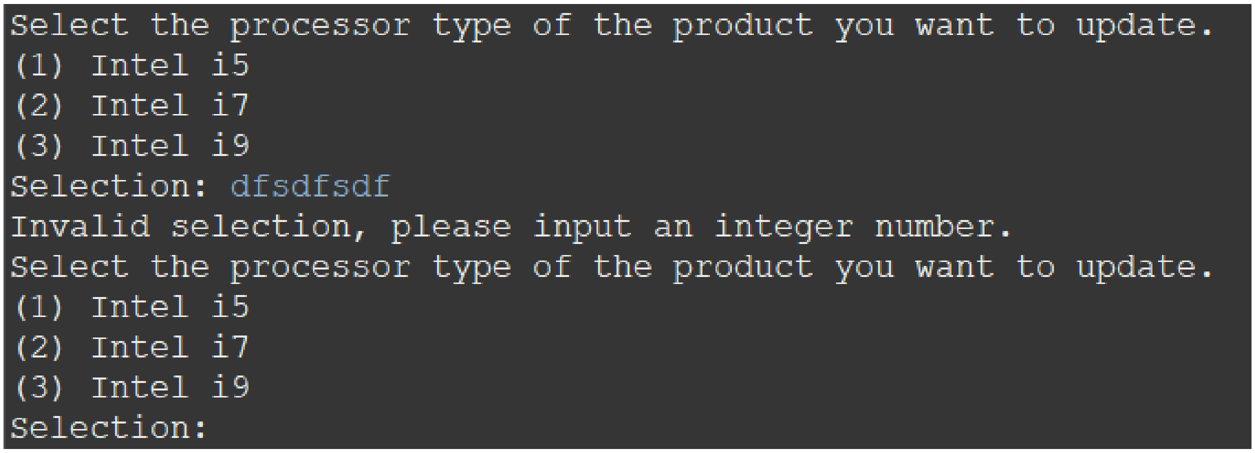
```
Select the new manufacturing country:
(1) Malaysia
(2) Japan
(3) America
Selection: 2
There are 4 records with manufacturing country of Malaysia:
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Malaysia         Intel i5           320 GB                      1 TB                       10
M73201024       Malaysia         Intel i7           320 GB                      1 TB                       10
M55002048       Malaysia         Intel i5           500 GB                      2 GB                       10
M910244096      Malaysia         Intel i9           1 TB                        4 GB                       20
(1) Update All
(2) Update One
Selection: fsdfsdf
Invalid selection, please input an integer number.
Select the manufacturing country of the product you want to update.
(1) Malaysia
(2) Japan
(3) America
Selection:
```

Figure 51 Invalid Selection – Non-Numeric Character



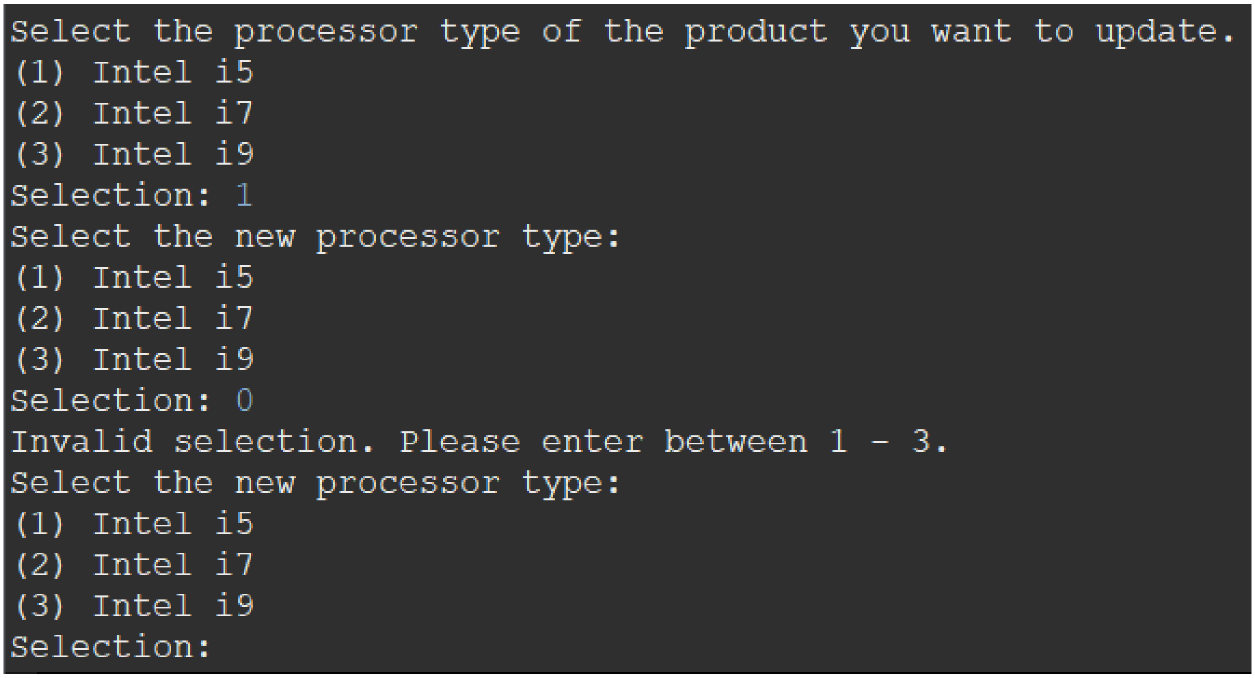
```
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection: 2
Select the processor type of the product you want to update.
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection: 0
Invalid selection. Please enter between 1 - 3.
Select the processor type of the product you want to update.
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection:
```

Figure 52 Invalid Selection – Not Within Selection Range



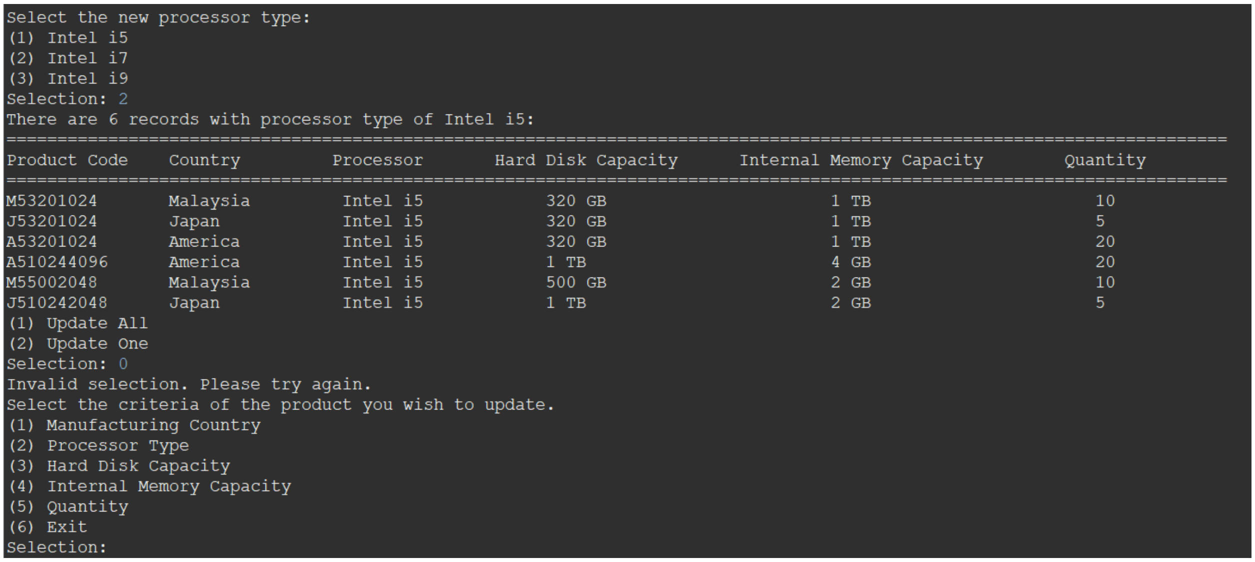
```
Select the processor type of the product you want to update.
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection: dfsdfsdf
Invalid selection, please input an integer number.
Select the processor type of the product you want to update.
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection:
```

Figure 53 Invalid Selection – Non-Numeric Character



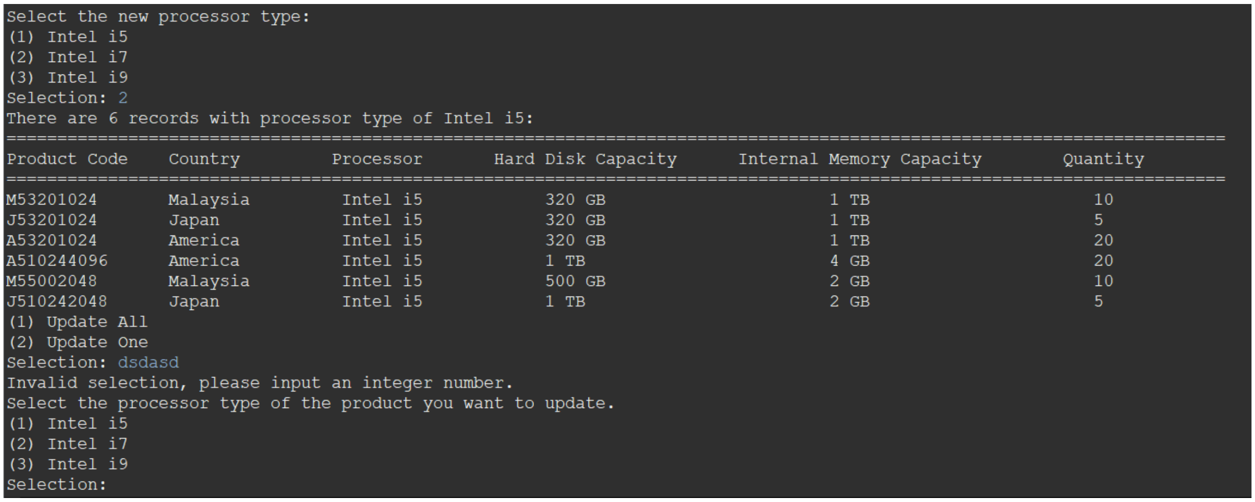
```
Select the processor type of the product you want to update.
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection: 1
Select the new processor type:
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection: 0
Invalid selection. Please enter between 1 - 3.
Select the new processor type:
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection:
```

Figure 54 Invalid Selection – Not Within Selection Range



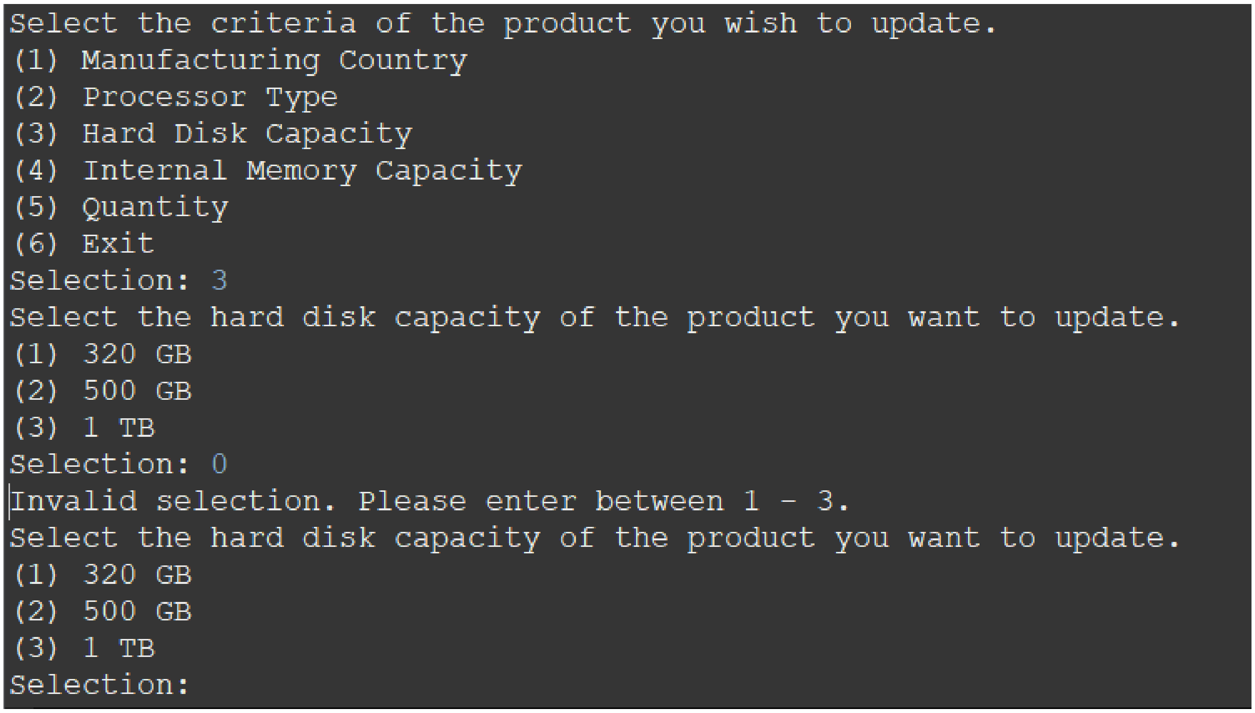
```
Select the new processor type:
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection: 2
There are 6 records with processor type of Intel i5:
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Malaysia         Intel i5           320 GB                      1 TB                       10
J53201024       Japan            Intel i5           320 GB                      1 TB                       5
A53201024       America          Intel i5           320 GB                      1 TB                       20
A510244096      America          Intel i5           1 TB                        4 GB                       20
M55002048       Malaysia         Intel i5           500 GB                      2 GB                       10
J510242048      Japan            Intel i5           1 TB                        2 GB                       5
(1) Update All
(2) Update One
Selection: 0
Invalid selection. Please try again.
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection:
```

Figure 55 Invalid Selection – Not Within Selection Range



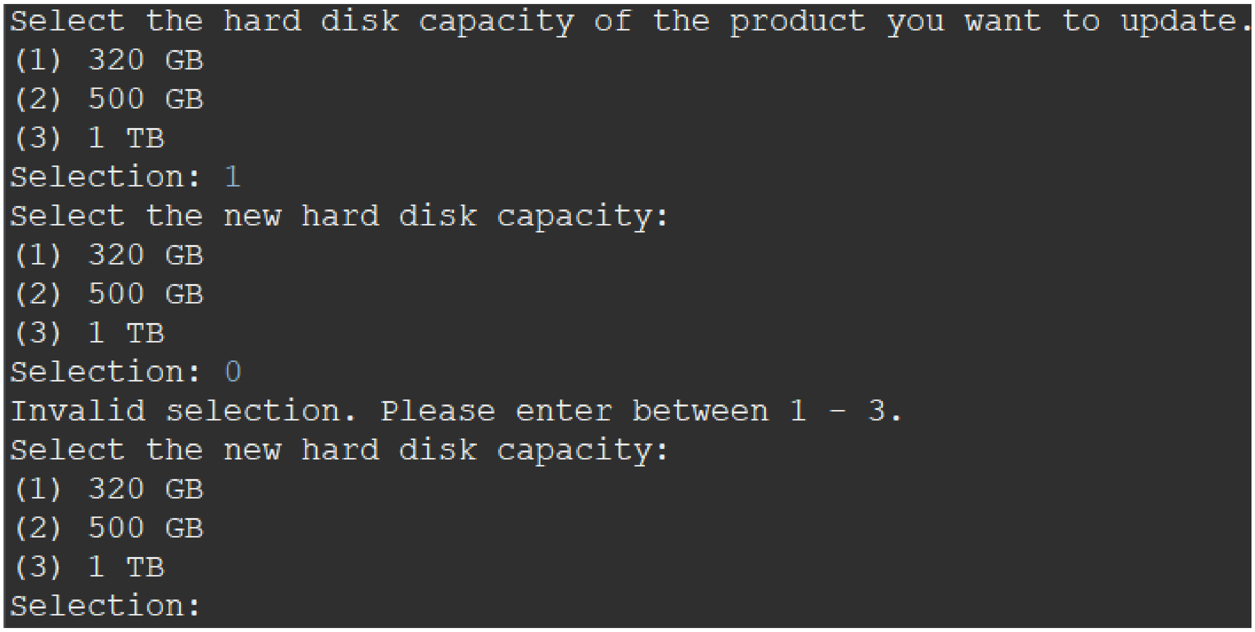
```
Select the new processor type:
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection: 2
There are 6 records with processor type of Intel i5:
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Malaysia         Intel i5           320 GB                      1 TB                        10
J53201024       Japan            Intel i5           320 GB                      1 TB                        5
A53201024       America          Intel i5           320 GB                      1 TB                        20
A510244096      America          Intel i5           1 TB                        4 GB                        20
M55002048       Malaysia         Intel i5           500 GB                      2 GB                        10
J510242048      Japan            Intel i5           1 TB                        2 GB                        5
(1) Update All
(2) Update One
Selection: dsdasd
Invalid selection, please input an integer number.
Select the processor type of the product you want to update.
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection:
```

Figure 56 Invalid Selection – Non-Numeric Character



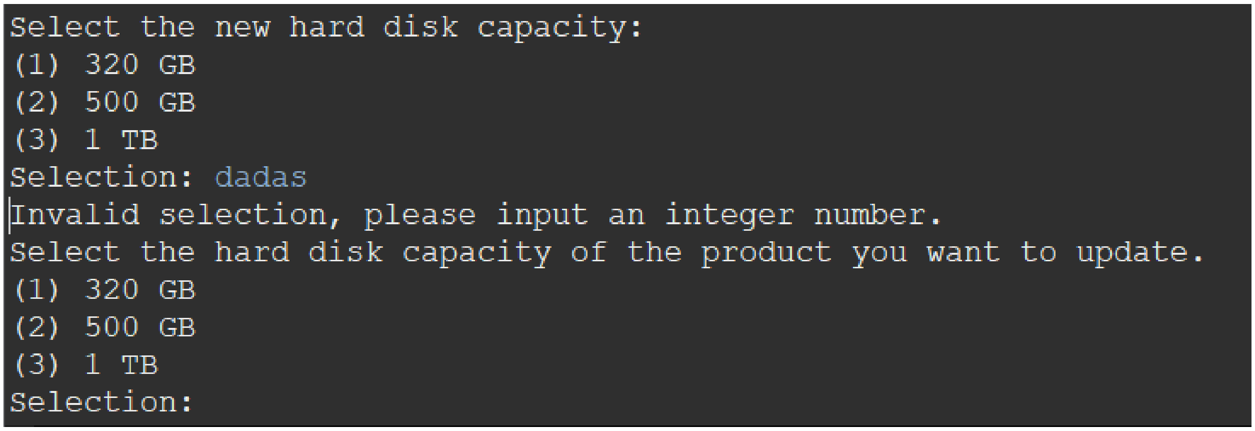
```
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection: 3
Select the hard disk capacity of the product you want to update.
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection: 0
Invalid selection. Please enter between 1 - 3.
Select the hard disk capacity of the product you want to update.
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection:
```

Figure 57 Invalid Selection – Not Within Selection Range



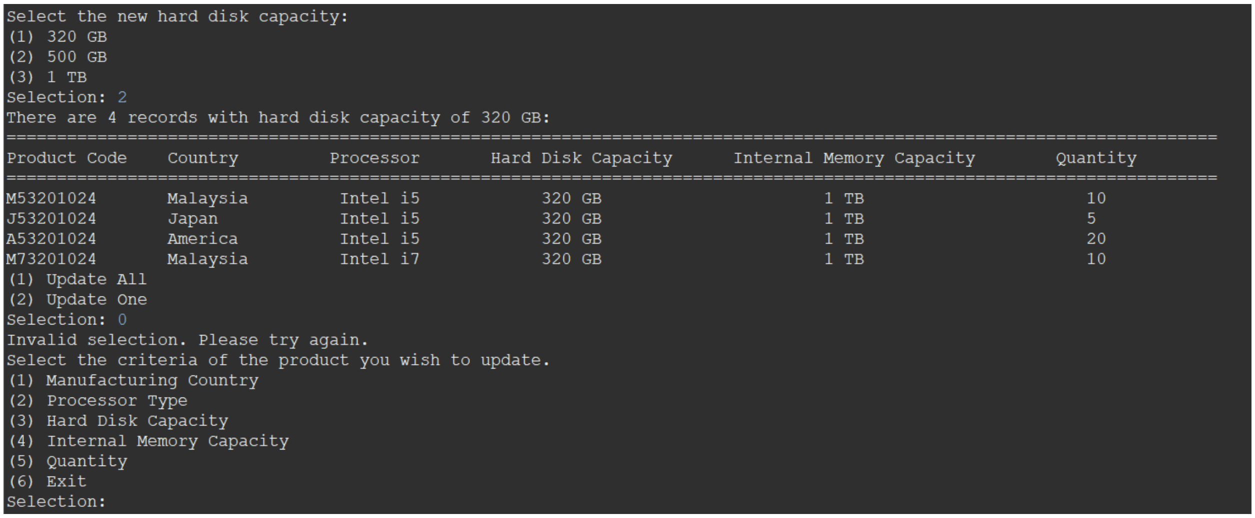
```
Select the hard disk capacity of the product you want to update.
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection: 1
Select the new hard disk capacity:
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection: 0
Invalid selection. Please enter between 1 - 3.
Select the new hard disk capacity:
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection:
```

Figure 58 Invalid Selection – Not Within Selection Range



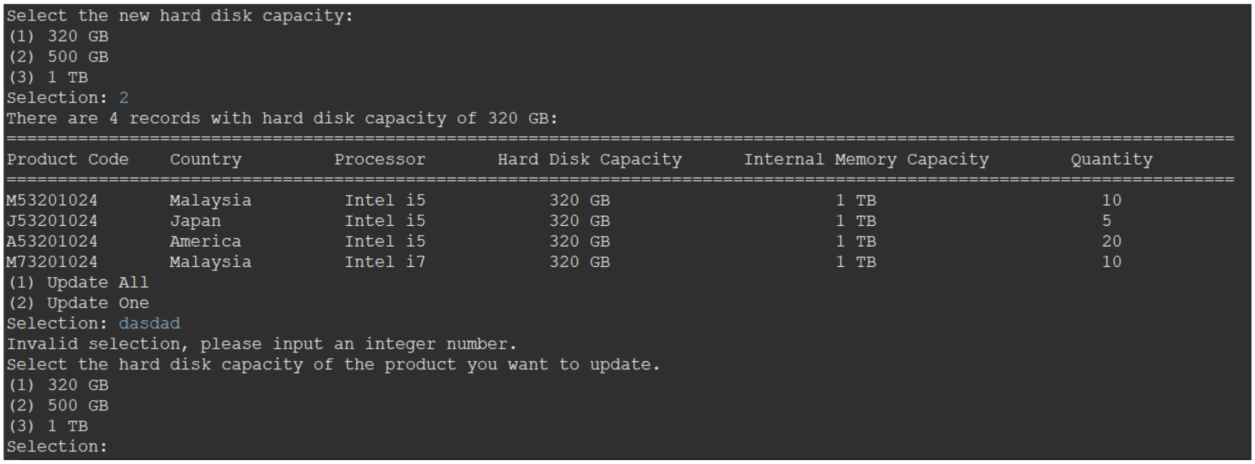
```
Select the new hard disk capacity:
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection: dadas
Invalid selection, please input an integer number.
Select the hard disk capacity of the product you want to update.
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection:
```

Figure 59 Invalid Selection – Non-Numeric Character



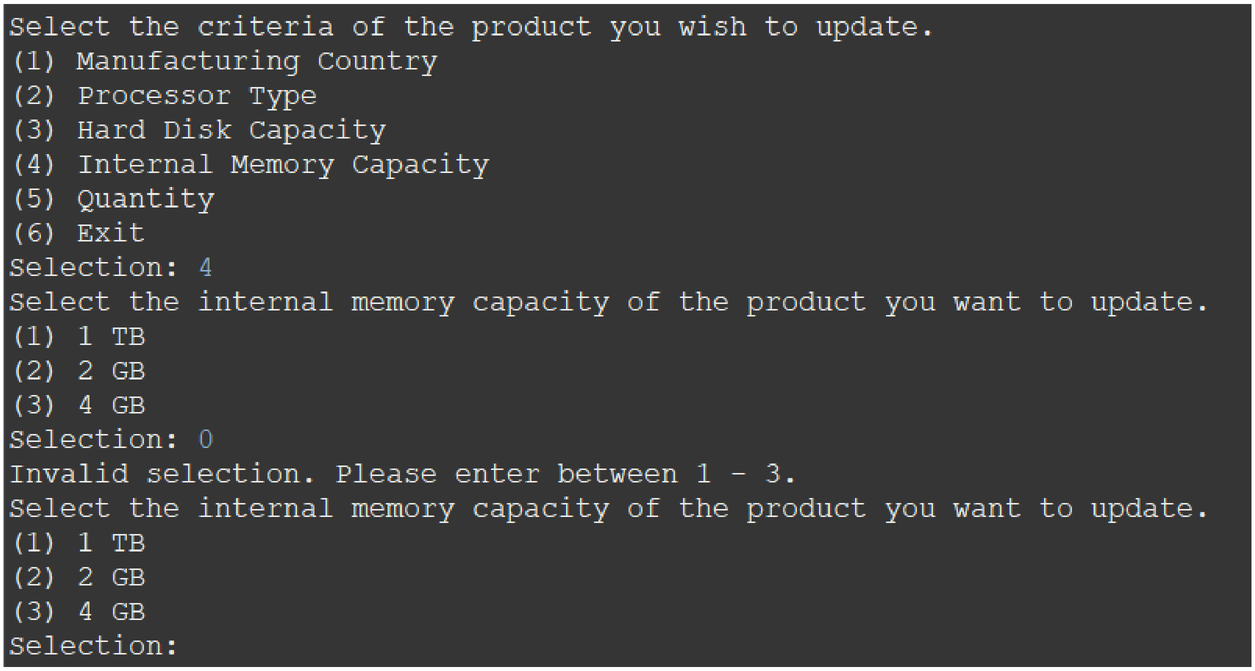
```
Select the new hard disk capacity:
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection: 2
There are 4 records with hard disk capacity of 320 GB:
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Malaysia         Intel i5            320 GB                      1 TB                      10
J53201024       Japan            Intel i5            320 GB                      1 TB                      5
A53201024       America          Intel i5            320 GB                      1 TB                      20
M73201024       Malaysia         Intel i7            320 GB                      1 TB                      10
(1) Update All
(2) Update One
Selection: 0
Invalid selection. Please try again.
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection:
```

Figure 60 Invalid Selection – Not Within Selection Range



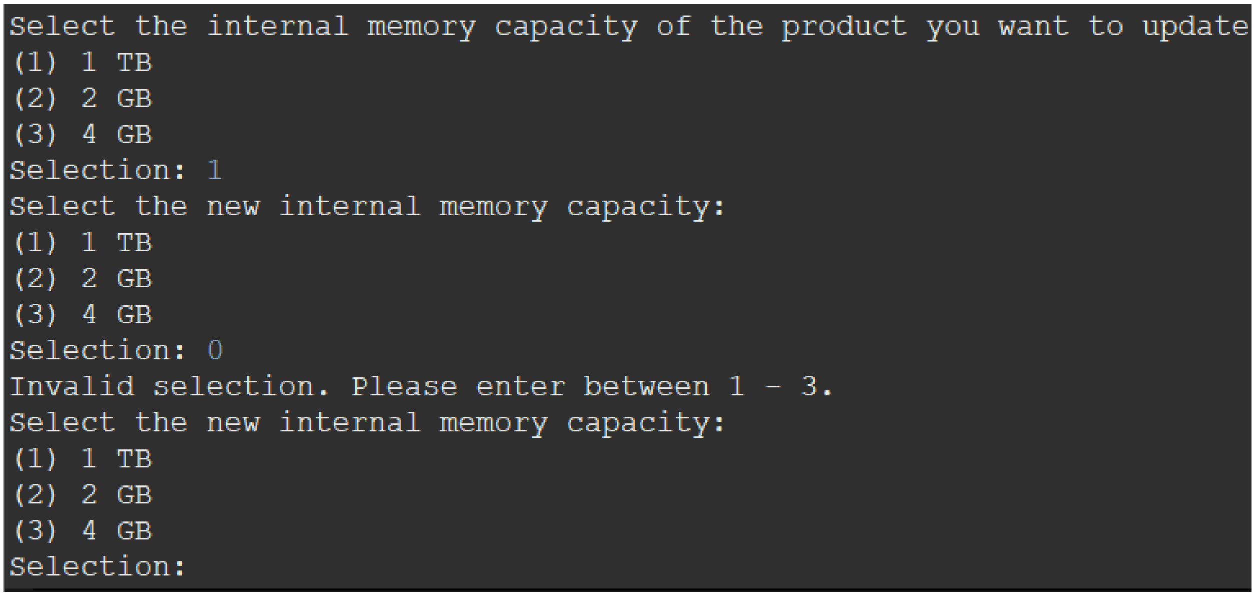
```
Select the new hard disk capacity:
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection: 2
There are 4 records with hard disk capacity of 320 GB:
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Malaysia         Intel i5            320 GB                      1 TB                      10
J53201024       Japan            Intel i5            320 GB                      1 TB                      5
A53201024       America          Intel i5            320 GB                      1 TB                      20
M73201024       Malaysia         Intel i7            320 GB                      1 TB                      10
(1) Update All
(2) Update One
Selection: dasdad
Invalid selection, please input an integer number.
Select the hard disk capacity of the product you want to update.
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection:
```

Figure 61 Invalid Selection – Non-Numeric Character



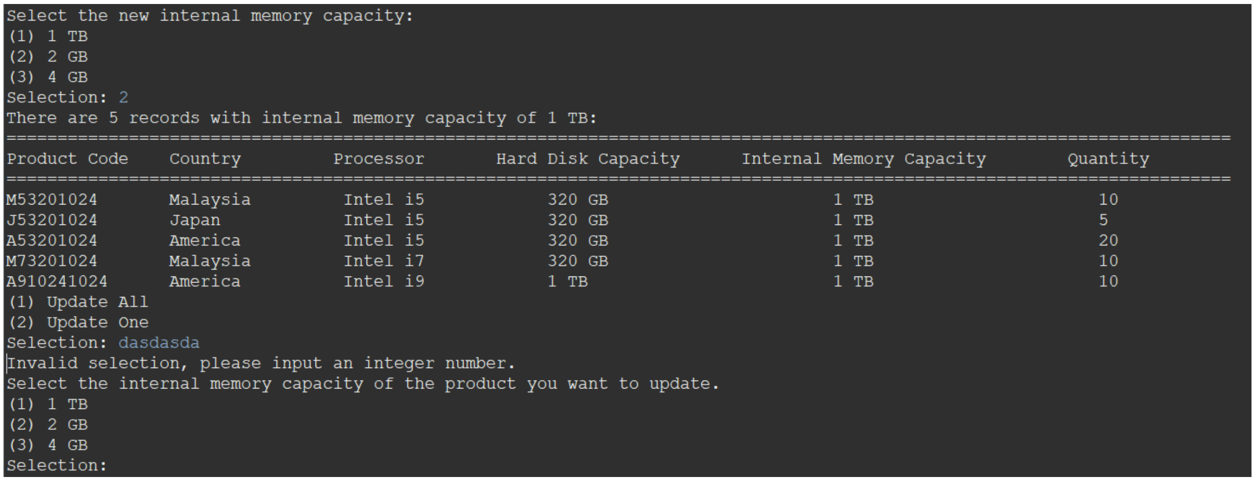
```
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection: 4
Select the internal memory capacity of the product you want to update.
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection: 0
Invalid selection. Please enter between 1 - 3.
Select the internal memory capacity of the product you want to update.
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection:
```

Figure 62 Invalid Selection – Not Within Selection Range



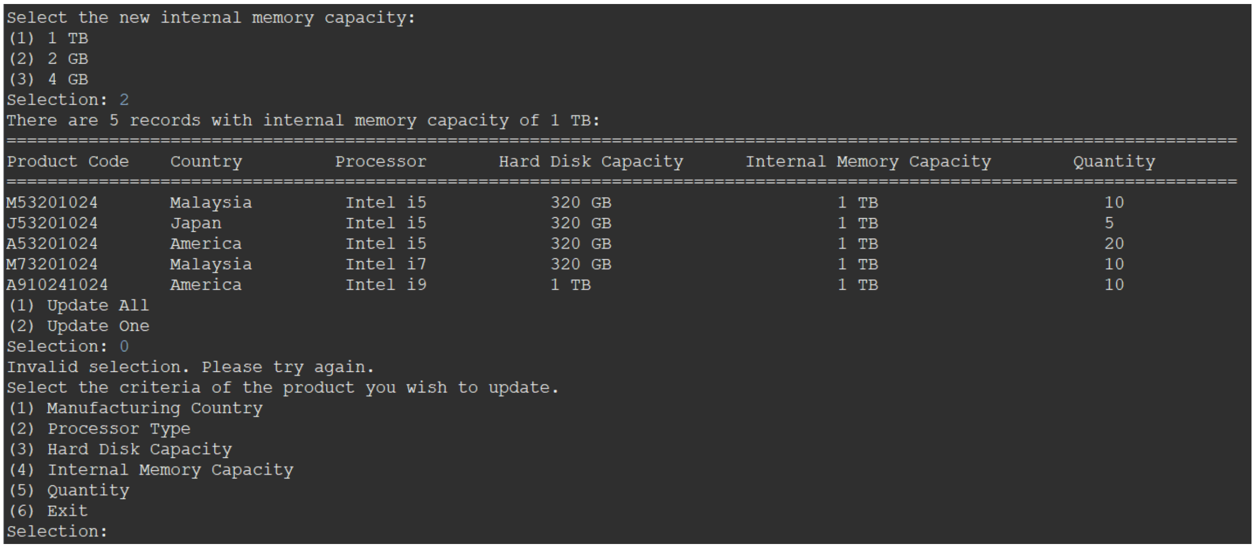
```
Select the internal memory capacity of the product you want to update
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection: 1
Select the new internal memory capacity:
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection: 0
Invalid selection. Please enter between 1 - 3.
Select the new internal memory capacity:
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection:
```

Figure 63 Invalid Selection – Not Within Selection Range



```
Select the new internal memory capacity:
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection: 2
There are 5 records with internal memory capacity of 1 TB:
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Malaysia         Intel i5            320 GB                      1 TB                      10
J53201024       Japan            Intel i5            320 GB                      1 TB                      5
A53201024       America          Intel i5            320 GB                      1 TB                      20
M73201024       Malaysia         Intel i7            320 GB                      1 TB                      10
A910241024      America          Intel i9            1 TB                        1 TB                      10
(1) Update All
(2) Update One
Selection: dasdasda
Invalid selection, please input an integer number.
Select the internal memory capacity of the product you want to update.
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection:
```

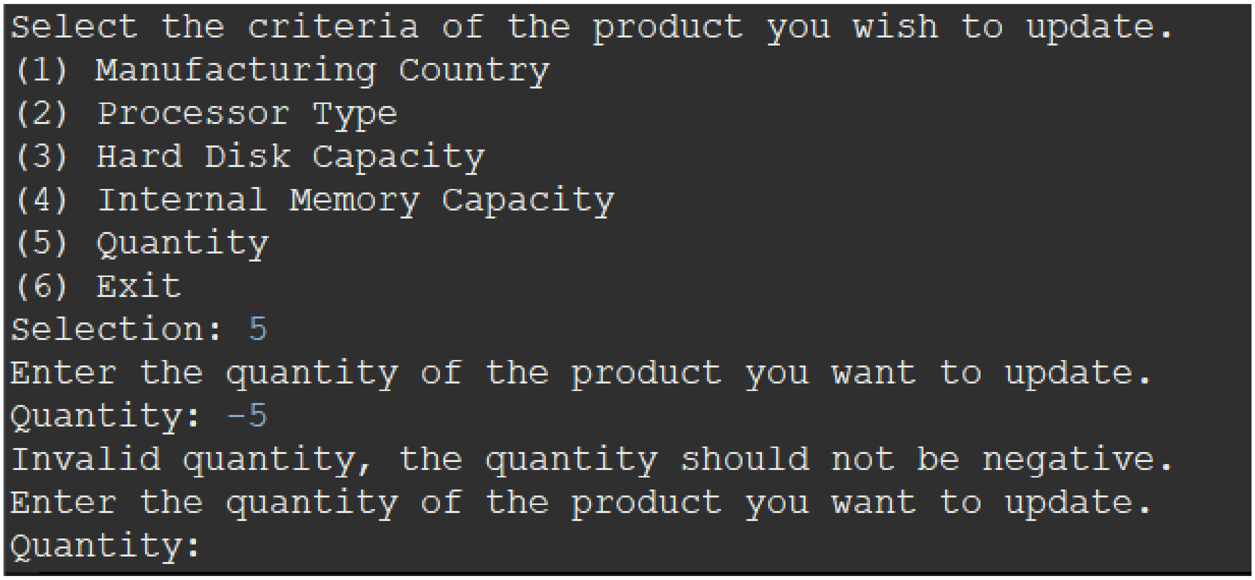
Figure 64 Invalid Selection – Non-Numeric Character



```
Select the new internal memory capacity:
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection: 2
There are 5 records with internal memory capacity of 1 TB:
========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
========================================================================================================================
M53201024       Malaysia         Intel i5            320 GB                      1 TB                      10
J53201024       Japan            Intel i5            320 GB                      1 TB                      5
A53201024       America          Intel i5            320 GB                      1 TB                      20
M73201024       Malaysia         Intel i7            320 GB                      1 TB                      10
A910241024      America          Intel i9            1 TB                        1 TB                      10
(1) Update All
(2) Update One
Selection: 0
Invalid selection. Please try again.
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection:
```

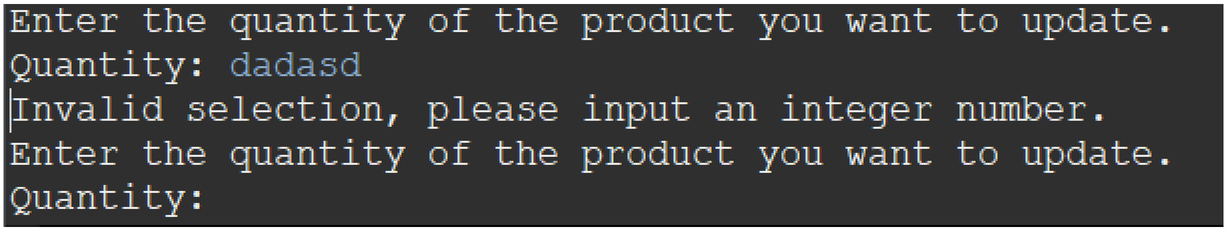
Figure 65 Invalid Selection – Not Within Selection Range

If user enter an invalid quantity which is contain non-numeric character, negative value or no records that match the selected quantity, the system will prompt out the error message and ask the user to enter the quantity again as shown in *Figure 66, Figure 67*, *Figure 68*.



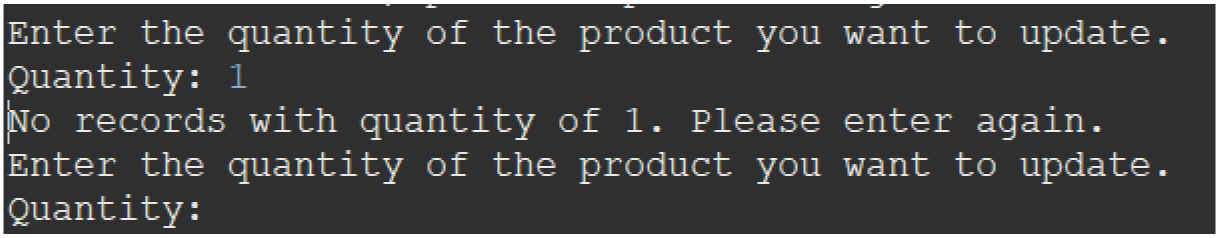
```
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection: 5
Enter the quantity of the product you want to update.
Quantity: -5
Invalid quantity, the quantity should not be negative.
Enter the quantity of the product you want to update.
Quantity:
```

Figure 66 Invalid Quantity – Negative Value



```
Enter the quantity of the product you want to update.
Quantity: dadasd
Invalid selection, please input an integer number.
Enter the quantity of the product you want to update.
Quantity:
```

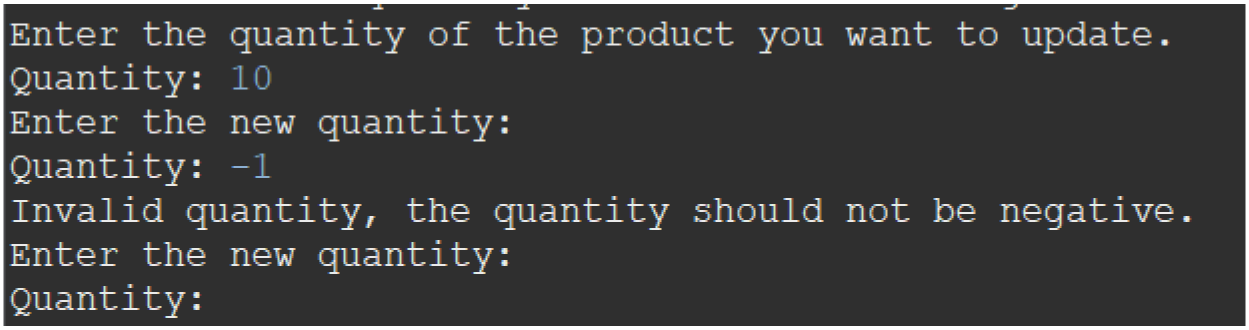
Figure 67 Invalid Quantity – Non-Numeric Character



```
Enter the quantity of the product you want to update.
Quantity: 1
No records with quantity of 1. Please enter again.
Enter the quantity of the product you want to update.
Quantity:
```

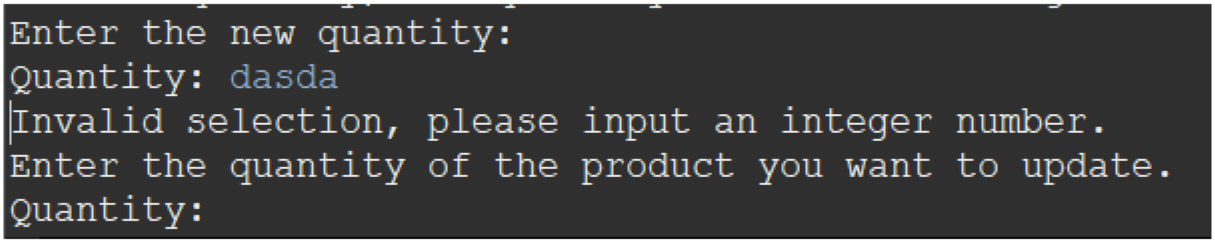
Figure 68 Invalid Quantity – No Records with Selected Quantity

If user enter an invalid new quantity which is contain non-numeric character or it is a negative value, the system will prompt out the error message and ask the user to enter the new quantity again as shown in *Figure 69*, *Figure 70*.



```
Enter the quantity of the product you want to update.
Quantity: 10
Enter the new quantity:
Quantity: -1
Invalid quantity, the quantity should not be negative.
Enter the new quantity:
Quantity:
```

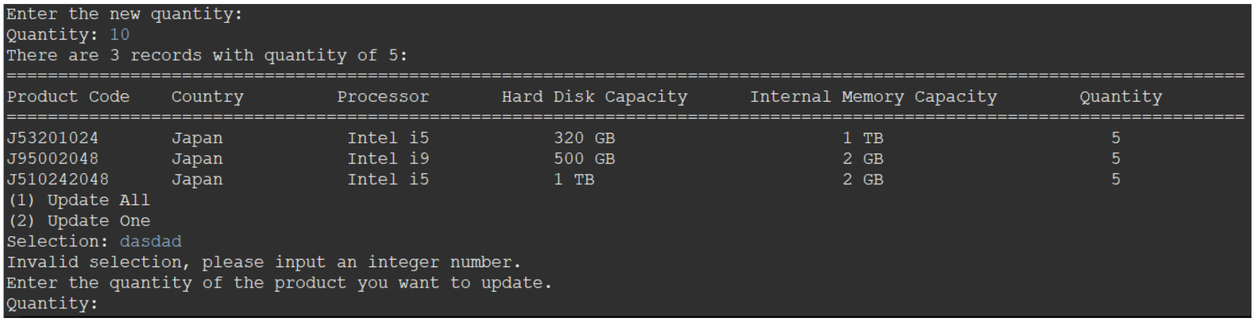
Figure 69 Invalid Quantity – Negative Value



```
Enter the new quantity:
Quantity: dasda
Invalid selection, please input an integer number.
Enter the quantity of the product you want to update.
Quantity:
```

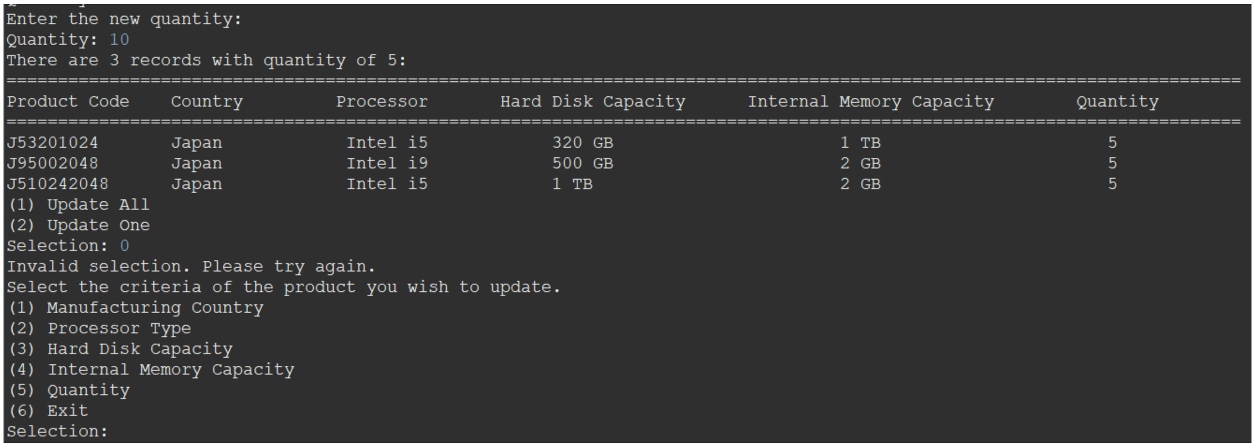
Figure 70 Invalid Quantity – Non-Numeric Character

If user enter an invalid selection which is contain non-numeric character or out of selection range, the system will prompt out the error message and ask the user to enter the selection again as shown in *Figure 71*, *Figure 72*.



```
Enter the new quantity:
Quantity: 10
There are 3 records with quantity of 5:
===========================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
===========================================================================================================================
J53201024       Japan            Intel i5           320 GB                      1 TB                        5
J95002048       Japan            Intel i9           500 GB                      2 GB                        5
J510242048      Japan            Intel i5           1 TB                        2 GB                        5
(1) Update All
(2) Update One
Selection: dasdad
Invalid selection, please input an integer number.
Enter the quantity of the product you want to update.
Quantity:
```

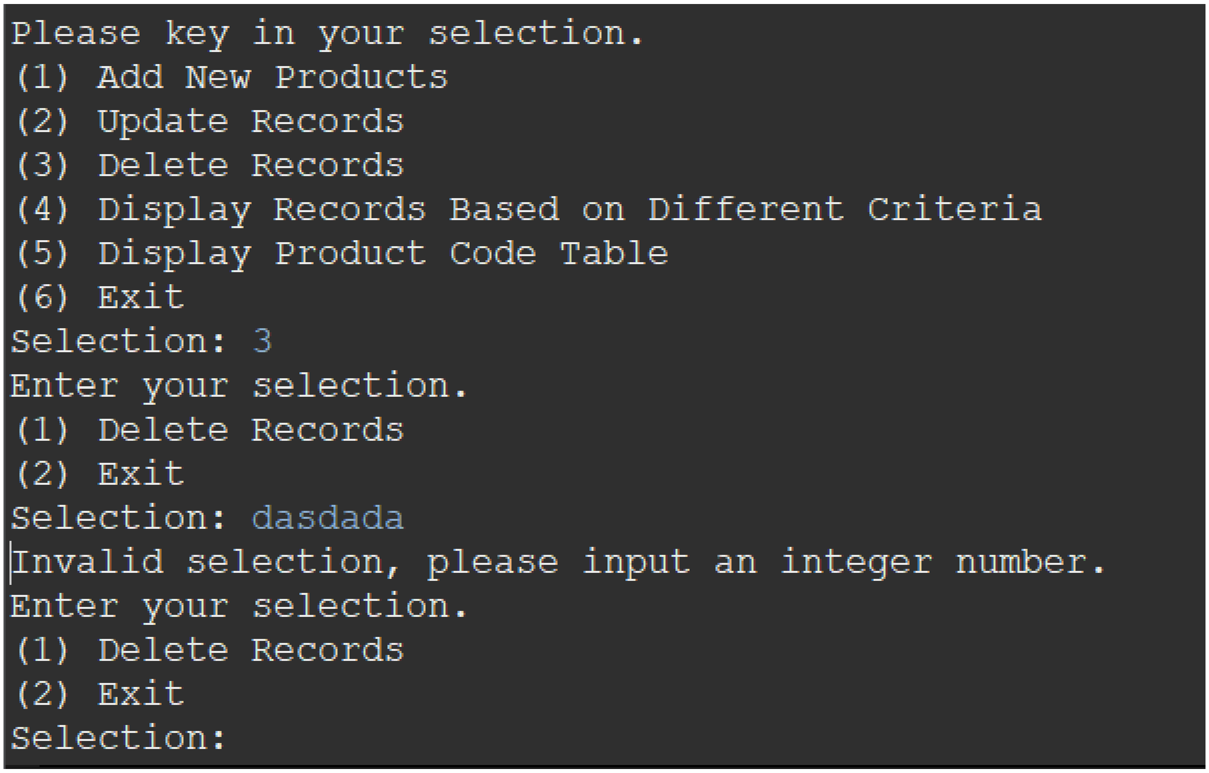
Figure 71 Invalid Selection – Non-Numeric Character



```
Enter the new quantity:
Quantity: 10
There are 3 records with quantity of 5:
=====================================================================================================================
Product Code    Country         Processor       Hard Disk Capacity      Internal Memory Capacity        Quantity
=====================================================================================================================
J53201024       Japan            Intel i5           320 GB                         1 TB                       5
J95002048       Japan            Intel i9           500 GB                         2 GB                       5
J510242048      Japan            Intel i5           1 TB                           2 GB                       5
(1) Update All
(2) Update One
Selection: 0
Invalid selection. Please try again.
Select the criteria of the product you wish to update.
(1) Manufacturing Country
(2) Processor Type
(3) Hard Disk Capacity
(4) Internal Memory Capacity
(5) Quantity
(6) Exit
Selection:
```

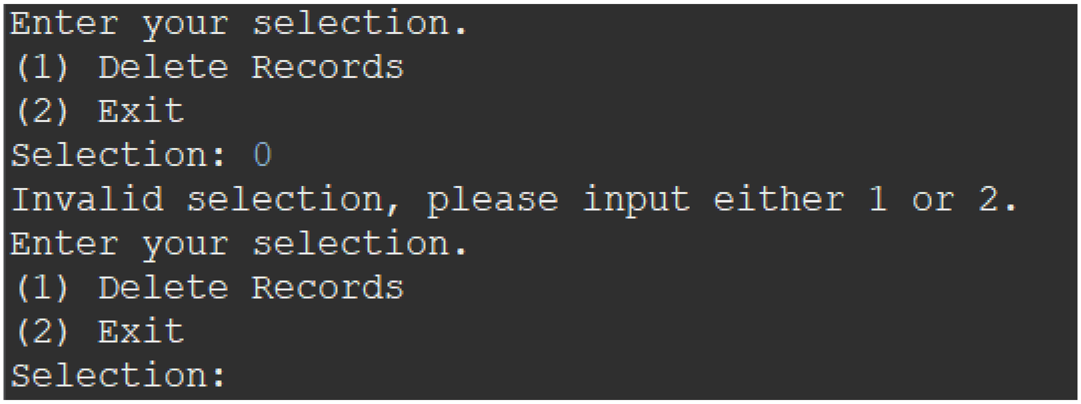
Figure 72 Invalid Selection – Not Within Selection Range

4. **(3) Delete Records Validation**
   If user enter an invalid selection which is contain non-numeric character or out of selection range, the system will prompt out the error message and ask the user to enter the selection again as shown in *Figure 73* to *Figure 77*.



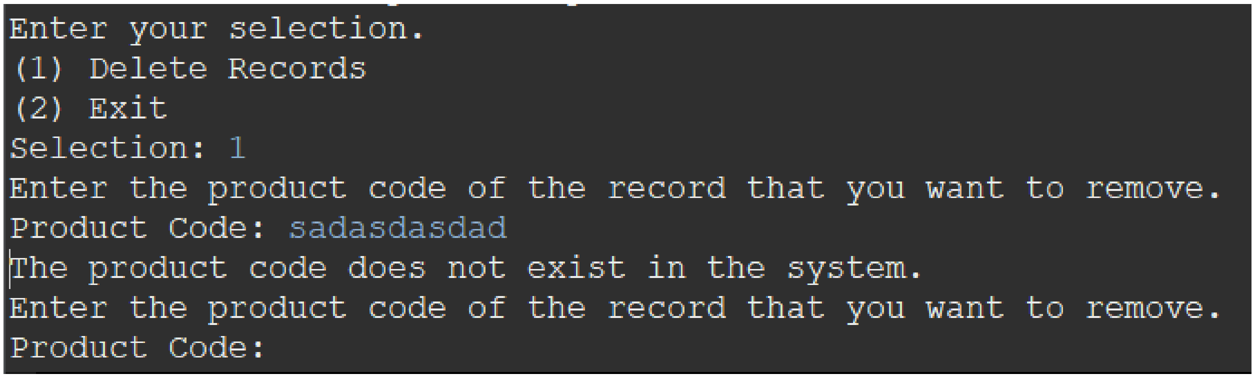
```
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 3
Enter your selection.
(1) Delete Records
(2) Exit
Selection: dasdada
Invalid selection, please input an integer number.
Enter your selection.
(1) Delete Records
(2) Exit
Selection:
```

Figure 73 Invalid Selection – Non-Numeric Character



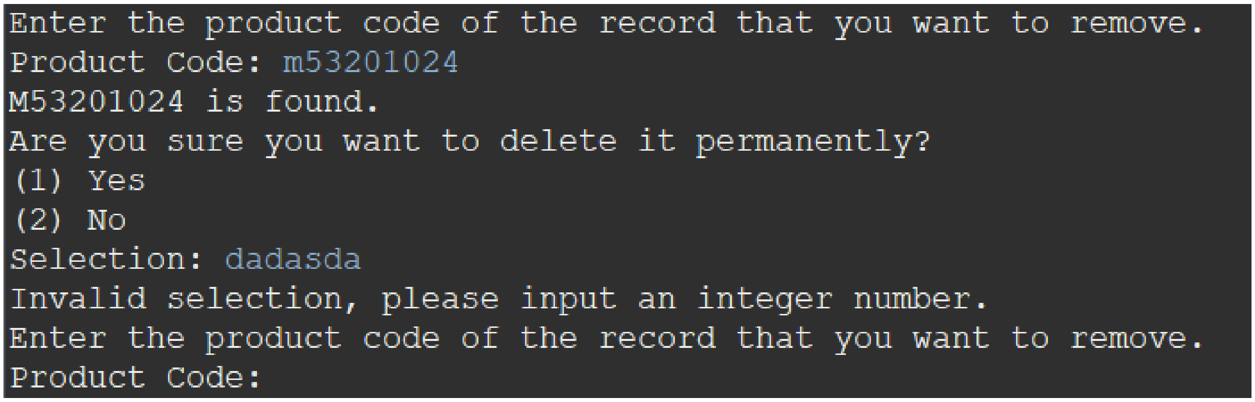
```
Enter your selection.
(1) Delete Records
(2) Exit
Selection: 0
Invalid selection, please input either 1 or 2.
Enter your selection.
(1) Delete Records
(2) Exit
Selection:
```

Figure 74 Invalid Selection – Not Within Selection Range



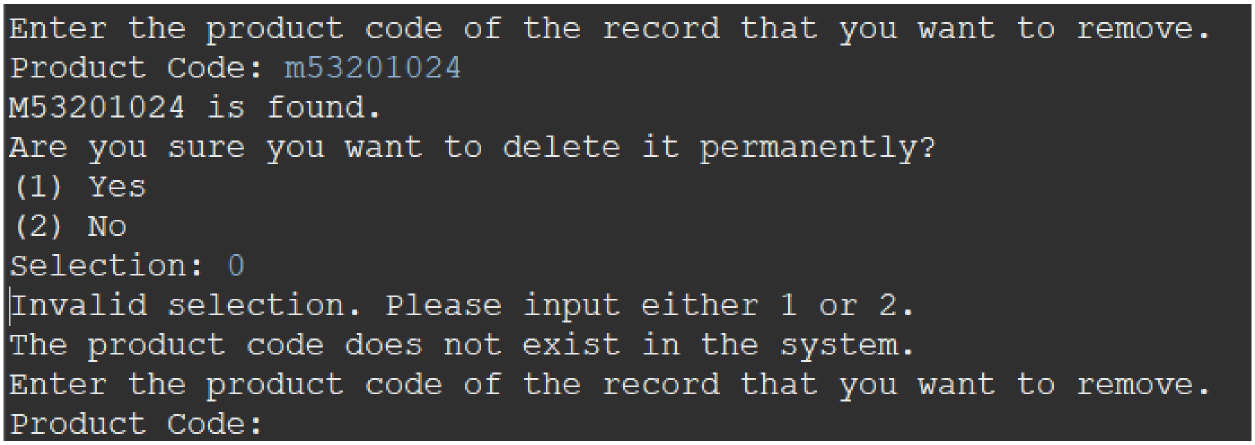
```
Enter your selection.
(1) Delete Records
(2) Exit
Selection: 1
Enter the product code of the record that you want to remove.
Product Code: sadasdasdad
The product code does not exist in the system.
Enter the product code of the record that you want to remove.
Product Code:
```

Figure 75 Invalid Selection – Non-Numeric Character



```
Enter the product code of the record that you want to remove.
Product Code: m53201024
M53201024 is found.
Are you sure you want to delete it permanently?
(1) Yes
(2) No
Selection: dadasda
Invalid selection, please input an integer number.
Enter the product code of the record that you want to remove.
Product Code:
```

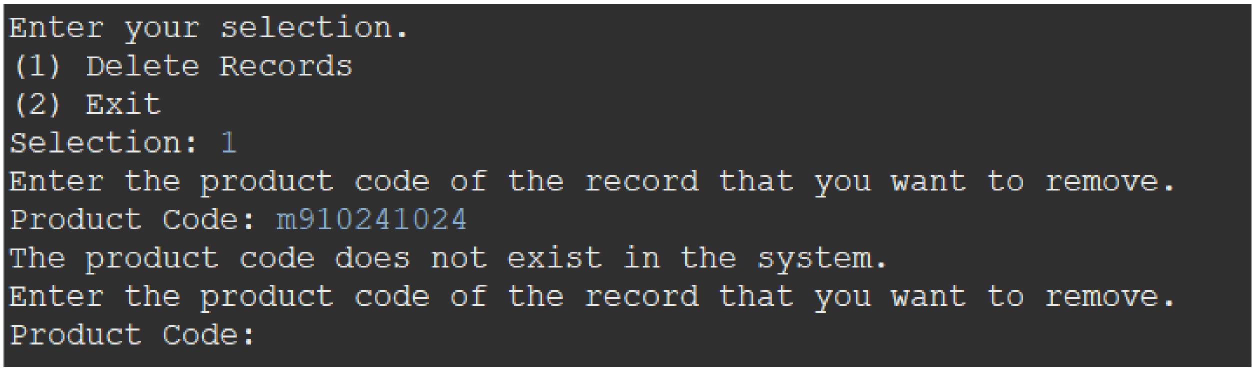
Figure 76 Invalid Selection – Non-Numeric Character



```
Enter the product code of the record that you want to remove.
Product Code: m53201024
M53201024 is found.
Are you sure you want to delete it permanently?
(1) Yes
(2) No
Selection: 0
Invalid selection. Please input either 1 or 2.
The product code does not exist in the system.
Enter the product code of the record that you want to remove.
Product Code:
```

Figure 77 Invalid Selection – Not Within Selection Range

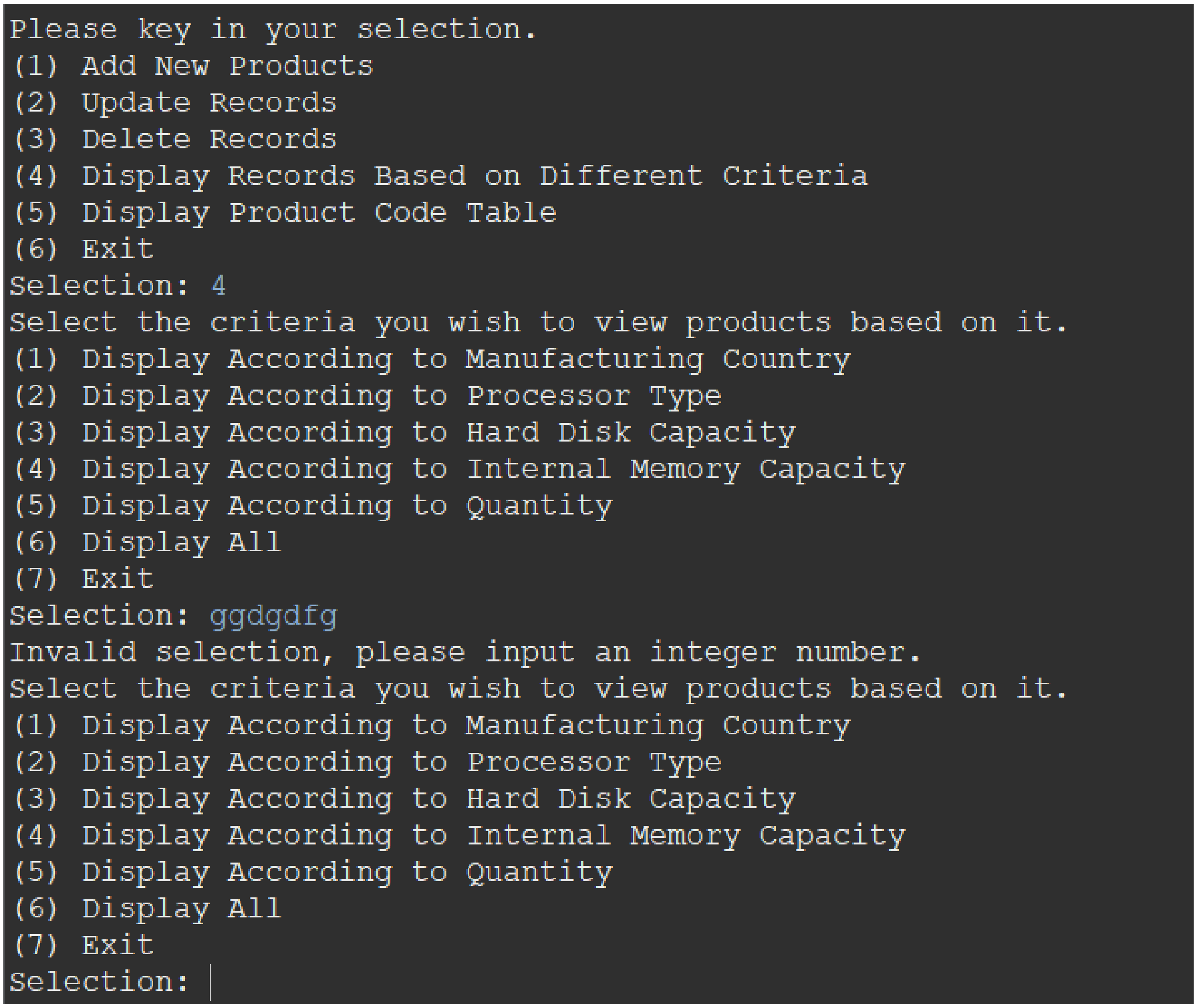
If user enter an invalid product code which is not exist in the system, the system will prompt out the error message and ask the user to enter the product code again as shown in *Figure 78*.



```
Enter your selection.
(1) Delete Records
(2) Exit
Selection: 1
Enter the product code of the record that you want to remove.
Product Code: m910241024
The product code does not exist in the system.
Enter the product code of the record that you want to remove.
Product Code:
```

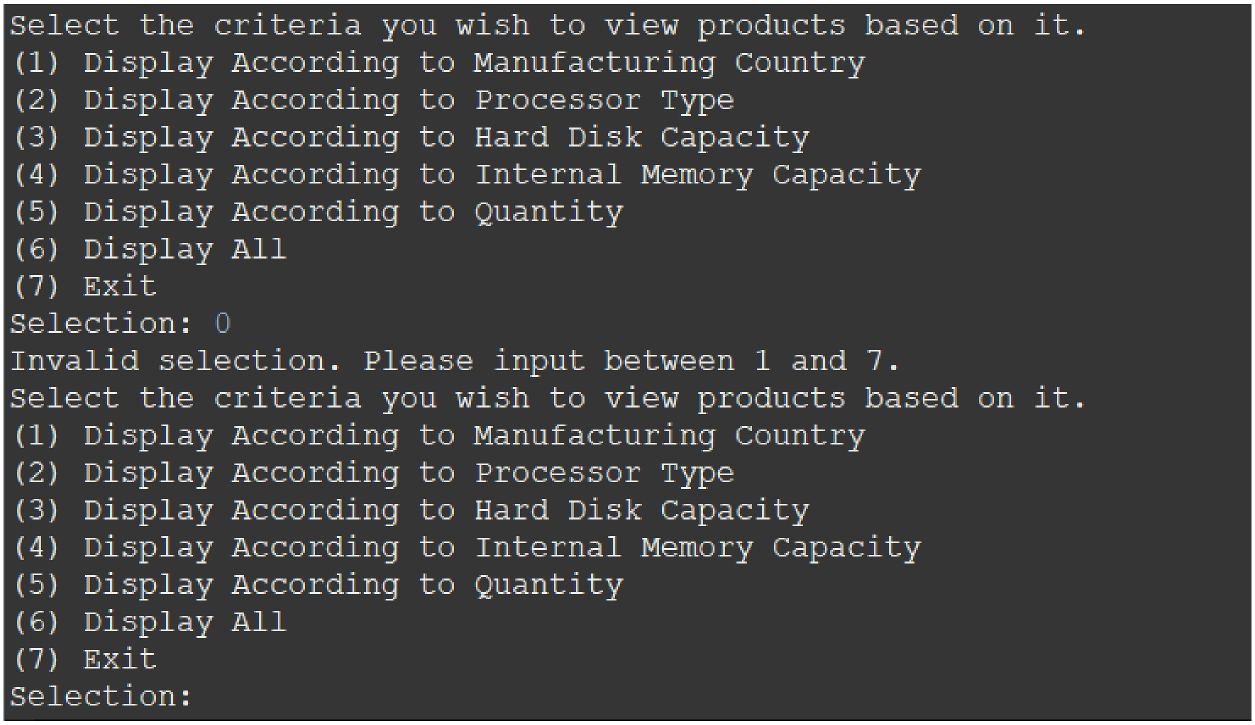
Figure 78 Invalid Product Code – Non-Existing Product Code

5. **(4) Display Records Based on Different Criteria Validation**
   If user enter an invalid selection which is contain non-numeric character or out of selection range, the system will prompt out the error message and ask the user to enter the selection again as shown in *Figure 79* to *Figure 88*.



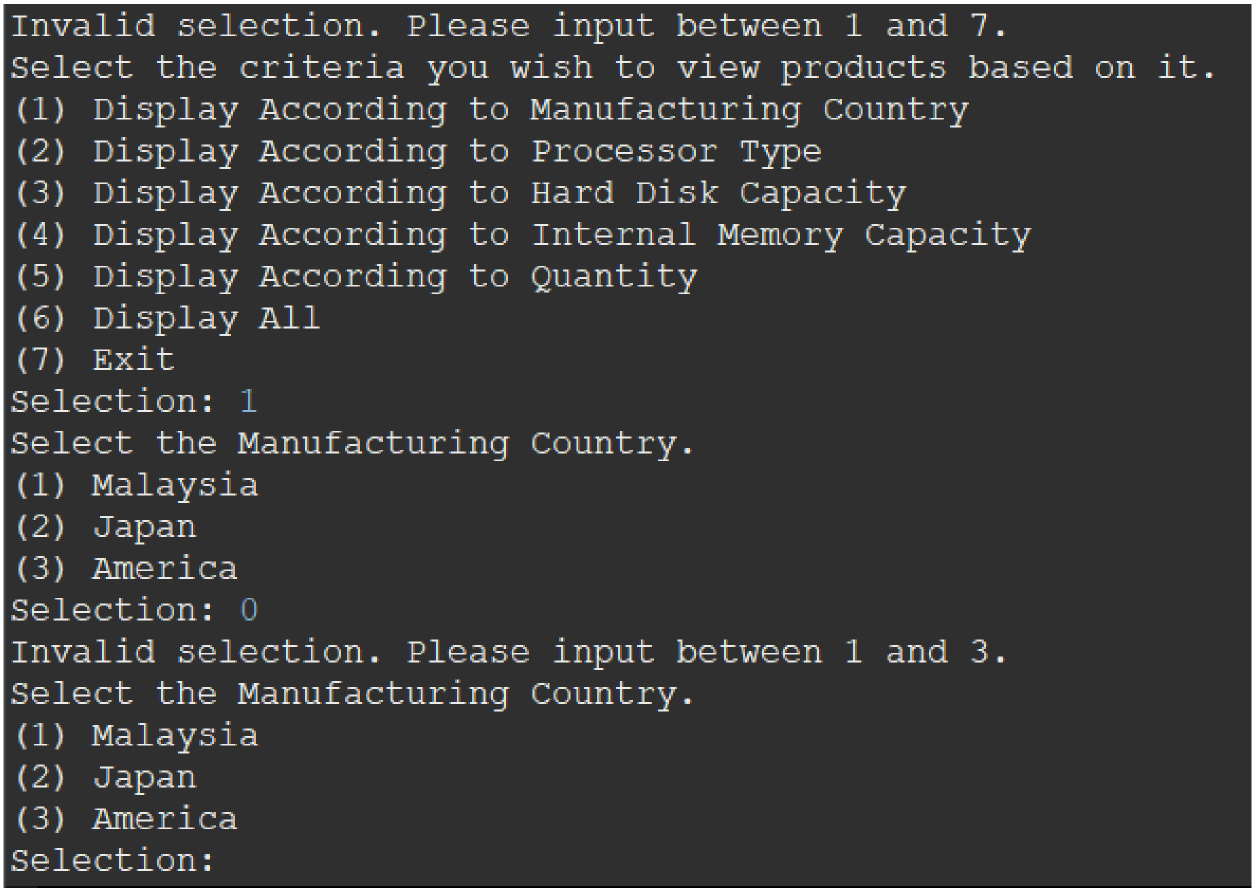
```
Please key in your selection.
(1) Add New Products
(2) Update Records
(3) Delete Records
(4) Display Records Based on Different Criteria
(5) Display Product Code Table
(6) Exit
Selection: 4
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection: ggdgdfg
Invalid selection, please input an integer number.
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

Figure 79 Invalid Selection – Non-Numeric Character



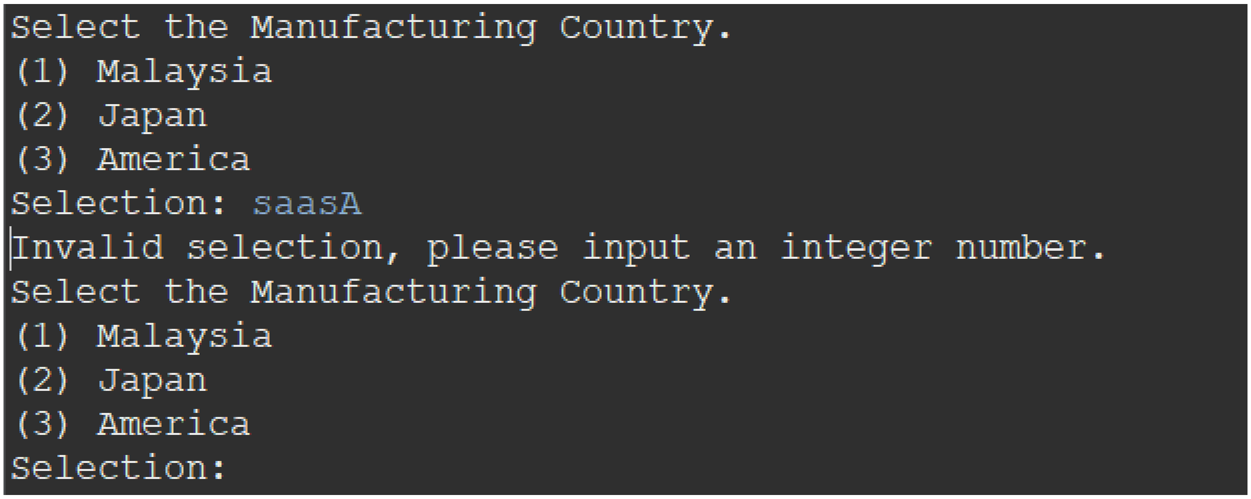
```
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection: 0
Invalid selection. Please input between 1 and 7.
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection:
```

Figure 80 Invalid Selection – Not Within Selection Range



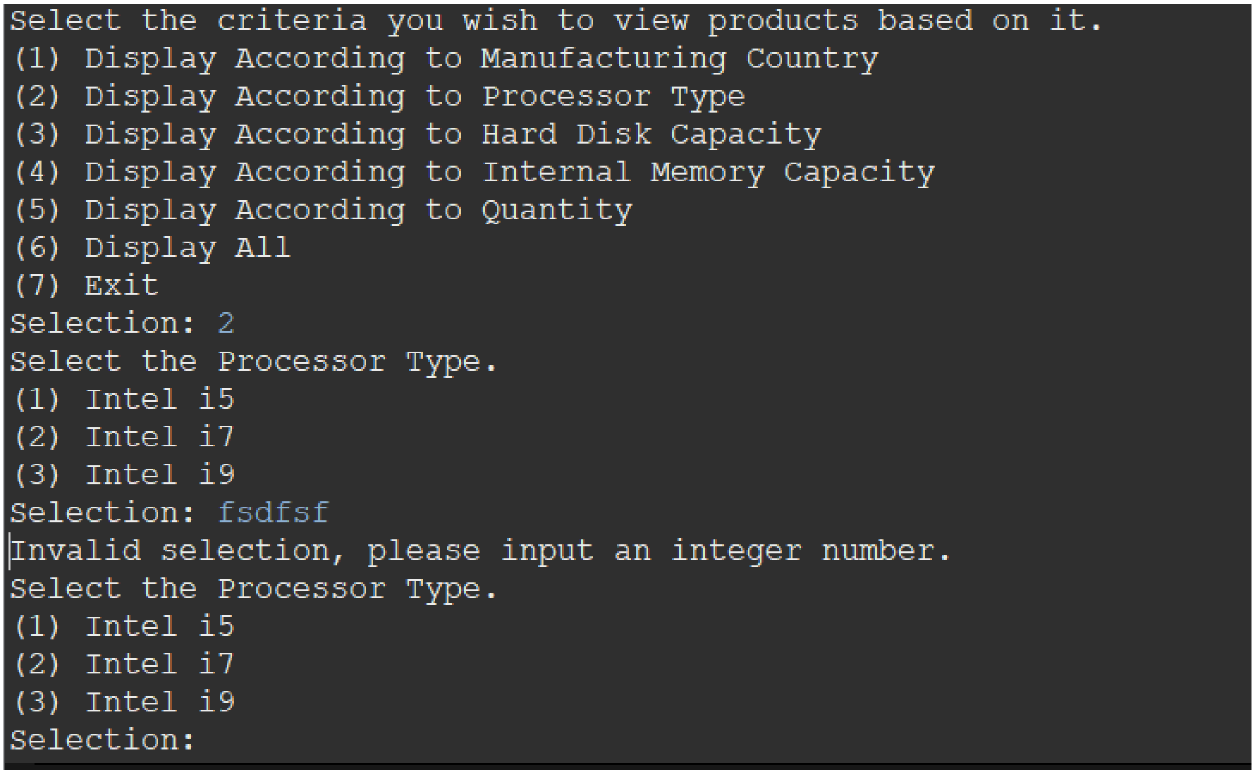
```
Invalid selection. Please input between 1 and 7.
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection: 1
Select the Manufacturing Country.
(1) Malaysia
(2) Japan
(3) America
Selection: 0
Invalid selection. Please input between 1 and 3.
Select the Manufacturing Country.
(1) Malaysia
(2) Japan
(3) America
Selection:
```

Figure 81 Invalid Selection – Not Within Selection Range



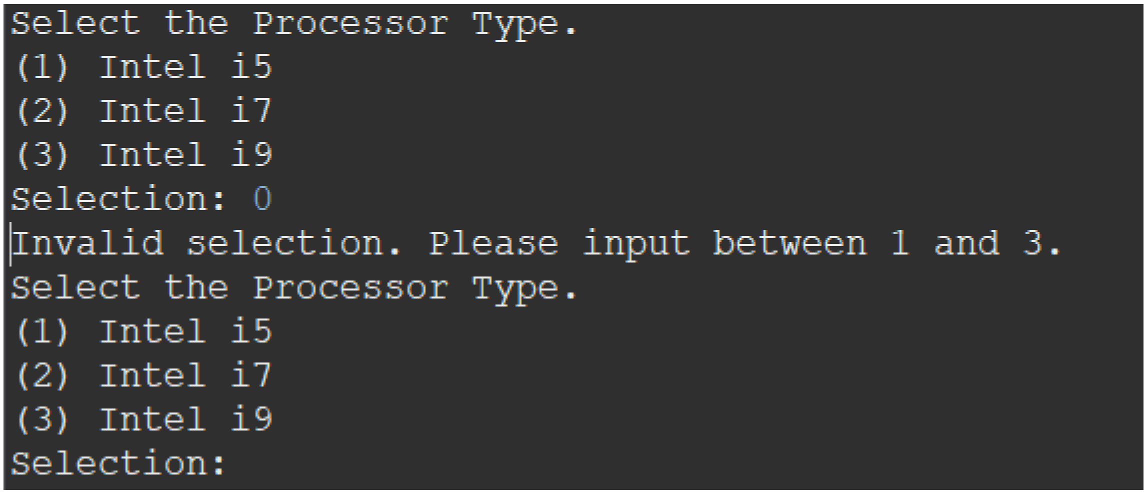
```
Select the Manufacturing Country.
(1) Malaysia
(2) Japan
(3) America
Selection: saasA
Invalid selection, please input an integer number.
Select the Manufacturing Country.
(1) Malaysia
(2) Japan
(3) America
Selection:
```

Figure 82 Invalid Selection – Non-Numeric Character



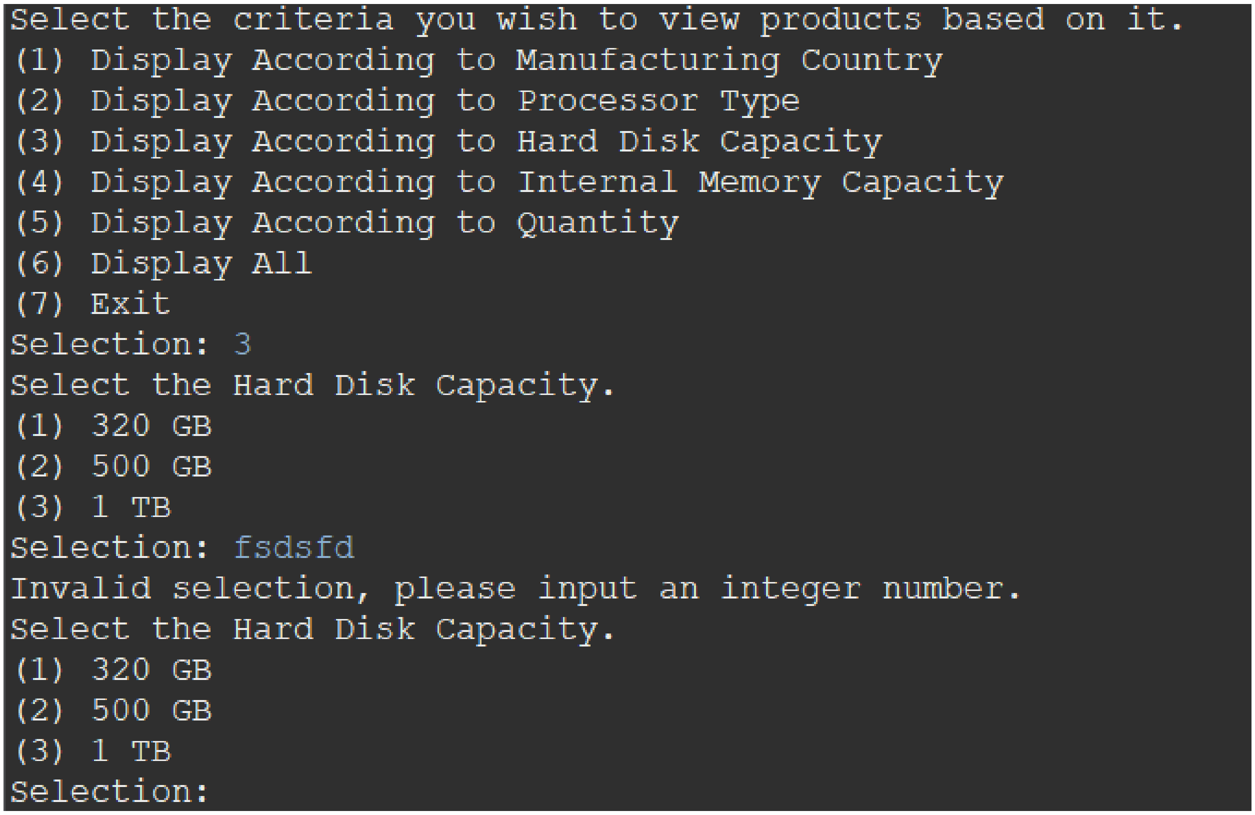
```
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection: 2
Select the Processor Type.
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection: fsdfsf
Invalid selection, please input an integer number.
Select the Processor Type.
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection:
```

Figure 83 Invalid Selection – Non-Numeric Character



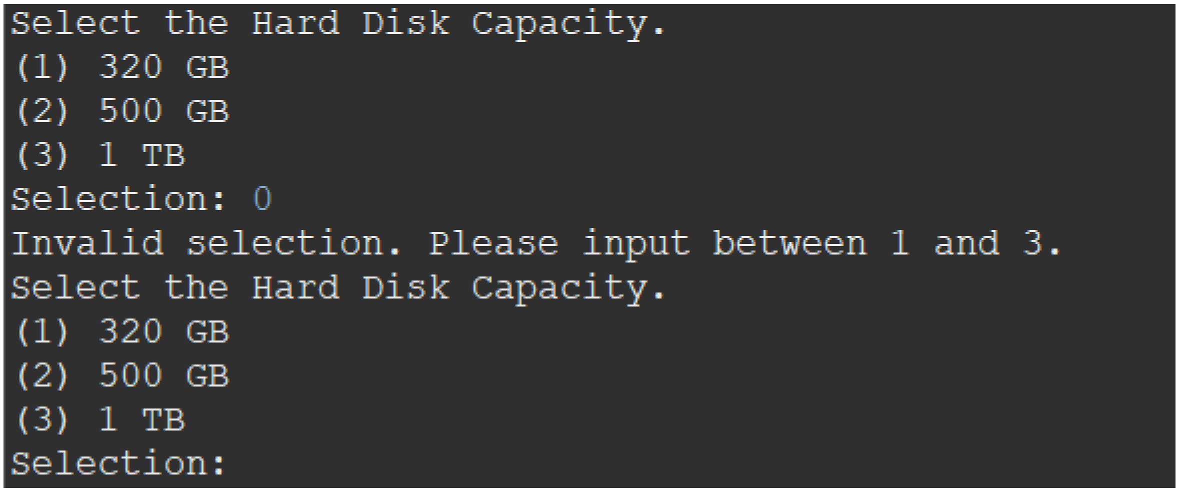
```
Select the Processor Type.
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection: 0
Invalid selection. Please input between 1 and 3.
Select the Processor Type.
(1) Intel i5
(2) Intel i7
(3) Intel i9
Selection:
```

Figure 84 Invalid Selection – Not Within Selection Range



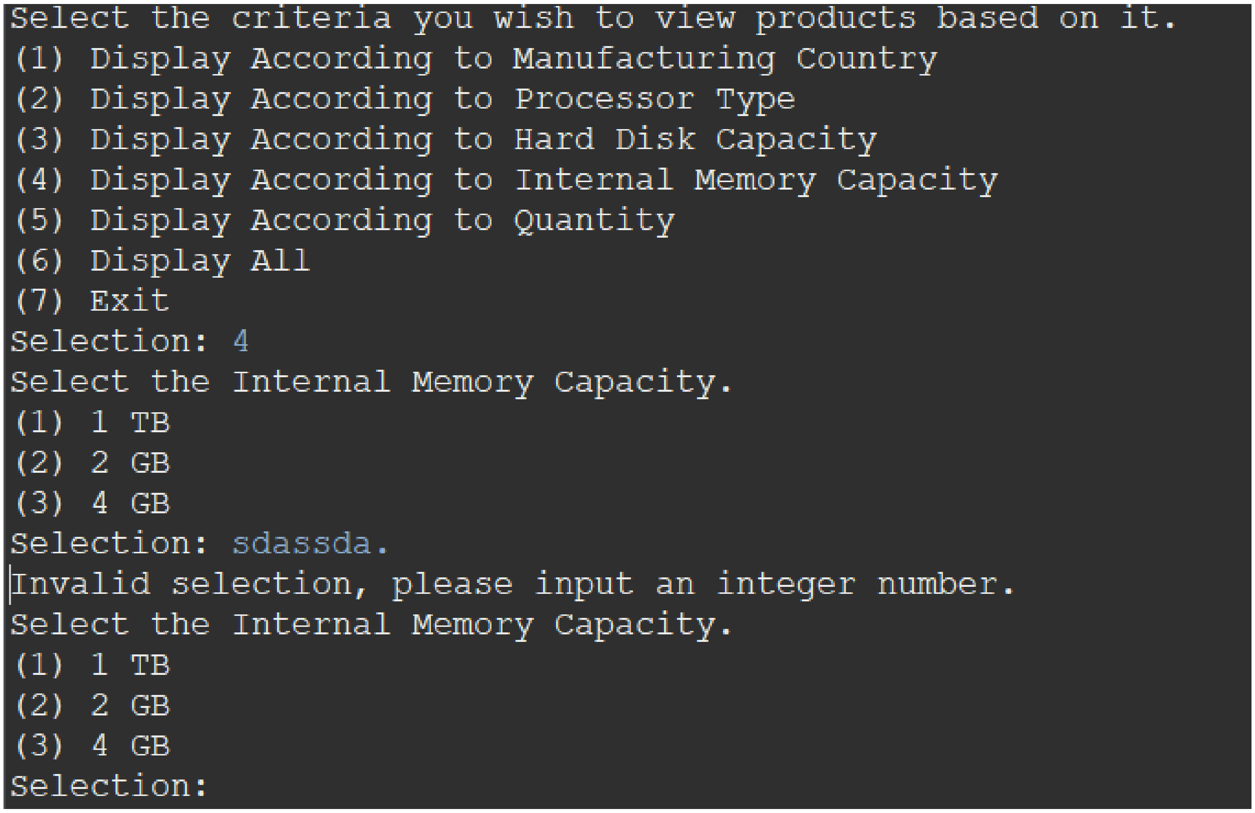
```
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection: 3
Select the Hard Disk Capacity.
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection: fsdsfd
Invalid selection, please input an integer number.
Select the Hard Disk Capacity.
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection:
```

Figure 85 Invalid Selection – Non-Numeric Character



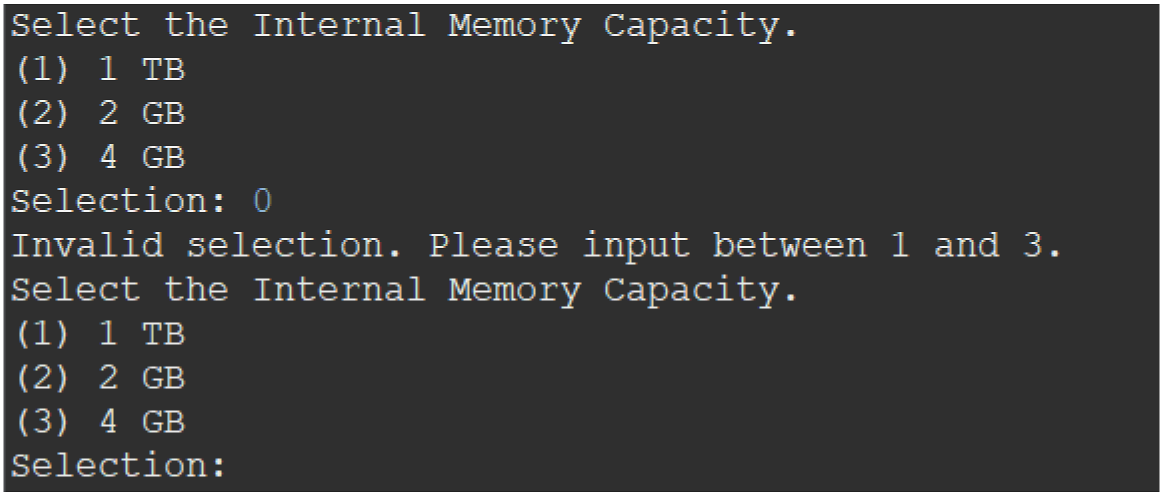
```
Select the Hard Disk Capacity.
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection: 0
Invalid selection. Please input between 1 and 3.
Select the Hard Disk Capacity.
(1) 320 GB
(2) 500 GB
(3) 1 TB
Selection:
```

Figure 86 Invalid Selection – Not Within Selection Range



```
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection: 4
Select the Internal Memory Capacity.
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection: sdassda.
Invalid selection, please input an integer number.
Select the Internal Memory Capacity.
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection:
```

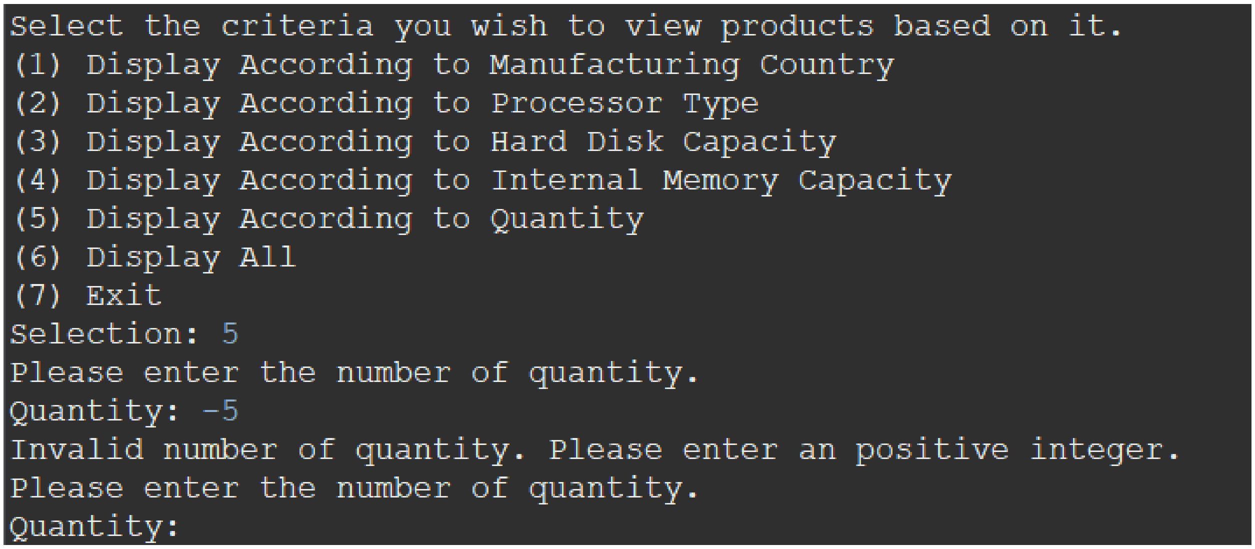
Figure 87 Invalid Selection – Non-Numeric Character



```
Select the Internal Memory Capacity.
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection: 0
Invalid selection. Please input between 1 and 3.
Select the Internal Memory Capacity.
(1) 1 TB
(2) 2 GB
(3) 4 GB
Selection:
```

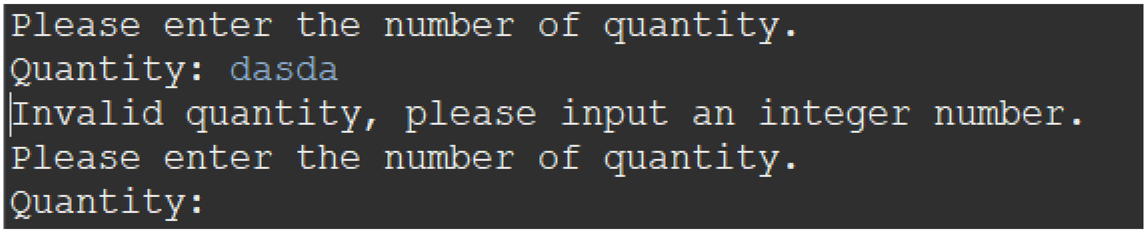
Figure 88 Invalid Selection – Not Within Selection Range

If user enter an invalid quantity which is contain non-numeric character or it is a negative value, the system will prompt out the error message and ask the user to enter the quantity again as shown in *Figure 89*, *Figure 90*.



```
Select the criteria you wish to view products based on it.
(1) Display According to Manufacturing Country
(2) Display According to Processor Type
(3) Display According to Hard Disk Capacity
(4) Display According to Internal Memory Capacity
(5) Display According to Quantity
(6) Display All
(7) Exit
Selection: 5
Please enter the number of quantity.
Quantity: -5
Invalid number of quantity. Please enter an positive integer.
Please enter the number of quantity.
Quantity:
```

Figure 89 Invalid Quantity – Negative Value



```
Please enter the number of quantity.
Quantity: dasda
Invalid quantity, please input an integer number.
Please enter the number of quantity.
Quantity:
```

Figure 90 Invalid Quantity – Non-Numeric Character

2.0 Individual Report - Chan Seow Fen (0207368)

Data structure refers to a specific format for data organization, processing, retrieval, and storage. Data structure allows users to quickly access and process the data they require. To illustrate it, there are two types of data structure, which is primitive data structure and non-primitive data structure (Loshin and Lewis, 2021). In short, primitive data structures is directly controlled by machine instructions while non-primitive data structures allow storing variable in multiple data type. The examples of primitive data structures are int, string, char, float and double. Non-primitive data type can be further divided into linear data structure and non-linear data structure. A linear data structure is made up of data elements that are organized in sequence, with each element connected to the elements before and after it while a non-linear data structure has no fixed order in which its components are connected, and each element can have multiple paths to other elements (Vishnu R, 2021). The examples of linear data structure are Array, ArrayList, Linked list, queue and stack while the examples of non-linear data structure are tree and graph. The data structure that has been chosen for developing the program is ArrayList. The ArrayList data structure is a resizable or dynamic array data structure that stores elements in sequential order and can be increased or reduced in size by adding or removing elements (Adservio.fr, 2023). One of the advantages of ArrayList are it does not require to mention the size when declaring the ArrayList. In addition, it benefits in the way that we can insert different types of variables into the ArrayList. Furthermore, elements can be added or removed from a particular position. Moreover, it can handle multiple elements that are null (pramodbablad, 2014). Additionally, using the ArrayList.get(element) method to get a particular element in it is extremely fast. On the other side, the disadvantage of ArrayList is slow insertion or deletion of data as updating the list need to shift the data. Furthermore, memory wastage occurred because larger components of a list require substantial contiguous blocks of memory. Moreover, resizing an ArrayList when it hits its initial capacity of 10 is a more expensive procedure because the elements are copied from the old to the new space with 50% more capacity (DevGlan, 2019).

ArrayList can be implemented by using ArrayList<String> syntax. Elements is inserted using add() method, accessed by using get() method and deleted by using remove() method. The application of ArrayList is used when need to store and manipulate large amounts of data especially when the size is not known beforehand. It has also been used when intend to insert duplicate elements into the list. Moreover, it is used when null elements needed to be inserted (Easy, 2020). Furthermore, it is also used to implement other data structure such as stack, queue and hash table. The memory used in ArrayList is totally depends on the size of the ArrayList as well as the data type stored in the ArrayList hence its space complexity is O(n), where the n is the number of elements in the list. In addition, the size of the ArrayList can be obtained using size() method while the capacity can be obtained using capacity() method. On the other hand, the time complexity of ArrayList varies depending on the specific operation being done. The time complexity is shown in the Table 1.2. For justification, operation 1 has an O(1) time complexity because ArrayList stores the elements in an array, and accessing an element by index is simply a matter of indexing into the array. Because all the elements after the specified index must be shifted to make space for the new element, operation 2 has an O(n) time complexity. The time complexity of operation 3 is O(1) on average, but O(n) in the worst situation. When the ArrayList is complete, it creates a new array that is twice the size of the previous array and copies its elements to the new array. As a result, this operation is an amortized O(1) process. Due to the reason that all the elements after the given index must be shifted to fill the gap left by the removed element, operation 4 has an O(n) time complexity. As it simply entails removing the last element from the array, operation 5 has a time complexity of O(1). Finally, because ArrayList have a built-in mechanism for looking for an element, the time complexity of Operation 6 is O(1).

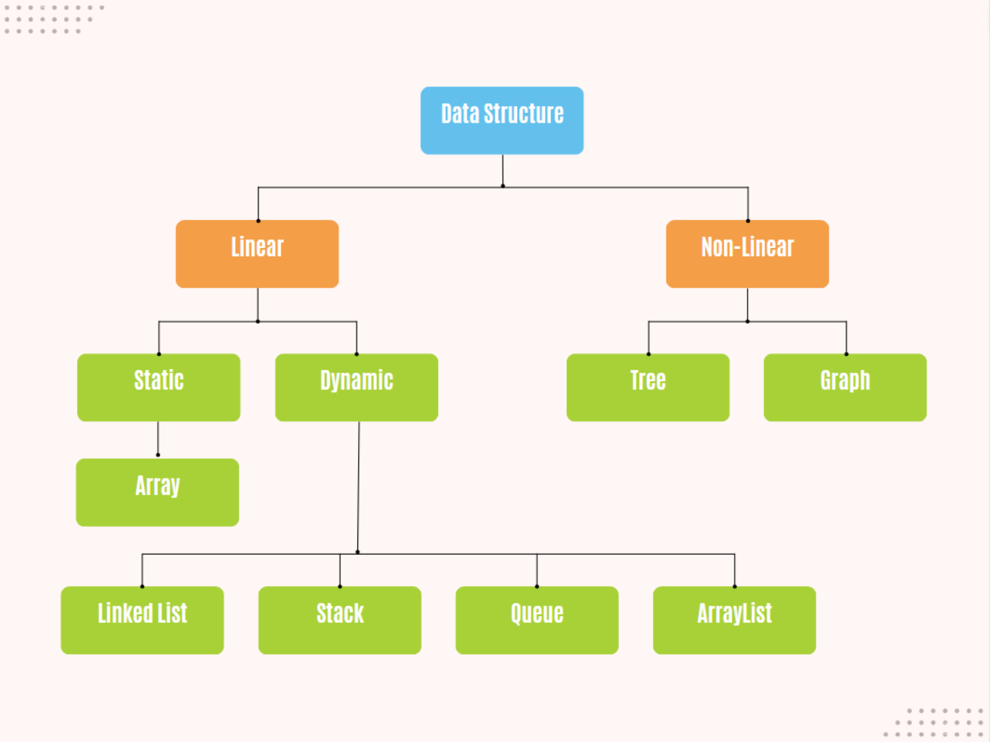


Figure 1.1 Type of Data Structure

	Operation	**Time Complexity in terms of Big-O notation**
1.	Accessing an element by index	O(1)
2.	Inserting an element at a specific index	O(n)
3.	Inserting an element at the end of the ArrayList	O(1) (amortized time)
4.	Removing an element from a specific index	O(n)
5.	Removing an element from the end of the ArrayList	O(1)
6.	Searching for an element	O(1)

Table 1.2 Time Complexity of ArrayList

3.0 Reference List

Adservio.fr. (2023). *ArrayList vs. LinkedList in Java: What I Need to Know*. [online] Available at: https://www.adservio.fr/post/arraylist-vs-linkedlist-in-java-what-i-need-to-know [Accessed 21 Mar. 2023].

DevGlan (2019). *Pros and Cons of Collection Types in Java | DevGlan*. [online] devglan. Available at: https://www.devglan.com/corejava/pros-and-cons-collection-java [Accessed 21 Mar. 2023].

Easy, S. (2020). *Realtime Use of ArrayList in Java with Example*. [online] Scientech Easy. Available at: https://www.scientecheasy.com/2020/09/use-of-arraylist-in-java.html/ [Accessed 21 Mar. 2023].

Loshin, D. and Lewis, S. (2021). *data structures*. [online] Data Management. Available at: https://www.techtarget.com/searchdatamanagement/definition/data-structure [Accessed 21 Mar. 2023].

pramodbablad (2014). *Advantages Of Using ArrayList Over Arrays*. [online] Java Concept Of The Day. Available at: https://javaconceptoftheday.com/advantages-of-using-arraylist-over-arrays/ [Accessed 21 Mar. 2023].

Vishnu R (2021). *What is Data Structure: Need, Types & Classification*. [online] Great Learning Blog: Free Resources what Matters to shape your Career! Available at: https://www.mygreatlearning.com/blog/data-structure-tutorial-for-beginners/ [Accessed 21 Mar. 2023].