CH13 練習

鄭安翔

ansel_cheng@hotmail.com

練習1 例外處理

1. 建立 ExceptionPractice 專案
2. 建立主類別ExceptionMain
   - main()方法中加入下列程式碼

     System.out.println("Reading from file:" + args[0]);
   - 使用 try-catch, 避免發生 ArrayIndexOutOfBoundsException
   - 加入下列程式碼

     BufferedReader b = new BufferedReader(new FileReader(args[0]));

     String s = null;

     while((s = b.readLine()) != null) {

         System.out.println(s);

     }
3. 專案properties, categories run中, 輸入argument欄位
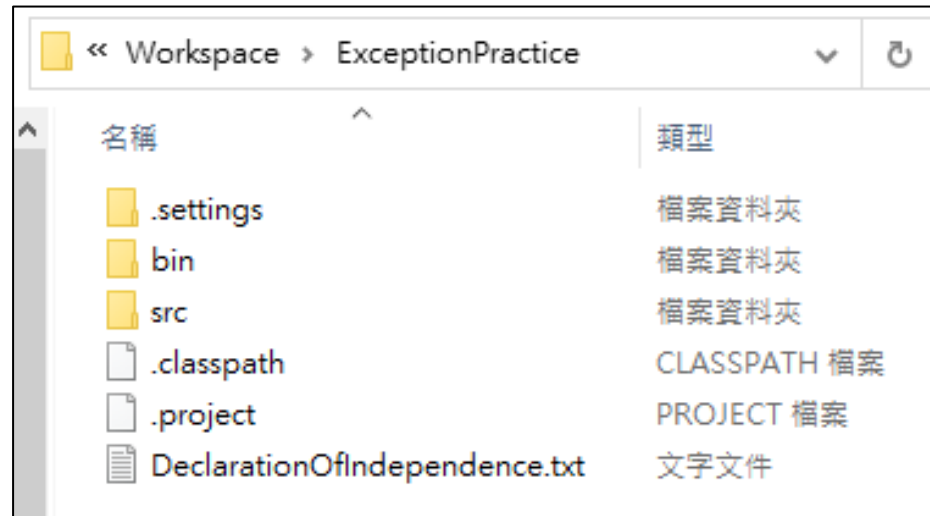   - DeclarationOfIndependence.txt 所在位置
4. 測試、執行

ExceptionMain

```java
ExceptionMain.java ×
1  import java.io.*;
2  public class ExceptionMain {
3
4      public static void main(String[] args){
5          try {
6              System.out.println("Reading from file:" + args[0]);
7          } catch (ArrayIndexOutOfBoundsException ex) {
8              System.err.println("未輸入讀取檔案名稱,程式結束");
9              System.exit(0);
10         }
11
12         try {
13             BufferedReader b = new BufferedReader(new FileReader(args[0]));
14             String s = null;
15             while ((s = b.readLine()) != null) {
16                 System.out.println(s);
17             }
18         } catch(FileNotFoundException ex) {
19             System.err.println("檔案不存在:"+args[0]);
20         } catch(IOException ex) {
21             System.err.println("檔案讀取失敗:"+ex.getMessage());
22         }
23     }
24
25 }
26
```

複製檔案置專案目錄

專案properties設定

測試、執行

<terminated> ExceptionMain [Java Application] C:\Program Files\Java\jdk-17.0.4\bin\javaw.exe (2024年3月18日 下午9:14:11 – 下午9:14:11) [pid: 20792]

```
Reading from file:DeclarationOfIndependence.txt
When, in the course of human events, it becomes necessary for one people to dissolve the political bands
which have connected them with another, and to assume among the powers of the earth, the separate and equal
station to which the laws of nature and of nature's God entitle them, a decent respect to the opinions of
mankind requires that they should declare the causes which impel them to the separation.

We hold these truths to be self-evident, that all men are created equal, that they are endowed by their
Creator with certain unalienable rights, that among these are life, liberty and the pursuit of happiness.
That to secure these rights, governments are instituted among men, deriving their just powers from the consent
of the governed. That whenever any form of government becomes destructive to these ends, it is the right of
the people to alter or to abolish it, and to institute new government, laying its foundation on such principles
and organizing its powers in such form, as to them shall seem most likely to effect their safety and happiness.
Prudence, indeed, will dictate that governments long established should not be changed for light and transient
causes; and accordingly all experience hath shown that mankind are more disposed to suffer, while evils are
sufferable, than to right themselves by abolishing the forms to which they are accustomed. But when a long
train of abuses and usurpations, pursuing invariably the same object evinces a design to reduce them under
absolute despotism, it is their right, it is their duty, to throw off such government, and to provide new
guards for their future security. --Such has been the patient sufferance of these colonies; and such is now
the necessity which constrains them to alter their former systems of government. The history of the present
King of Great Britain is a history of repeated injuries and usurpations, all having in direct object the
establishment of an absolute tyranny over these states. To prove this, let facts be submitted to a candid
world.

He has refused his assent to laws, the most wholesome and necessary for the public good.

He has forbidden his governors to pass laws of immediate and pressing importance, unless suspended in their
operation till his assent should be obtained; and when so suspended, he has utterly neglected to attend to them.

He has refused to pass other laws for the accommodation of large districts of people, unless those people would
relinquish the right of representation in the legislature, a right inestimable to them and formidable to tyrants only.

He has called together legislative bodies at places unusual, uncomfortable, and distant from the depository of
their public records, for the sole purpose of fatiguing them into compliance with his measures.

He has dissolved representative houses repeatedly, for opposing with manly firmness his invasions on the rights
of the people.

He has refused for a long time, after such dissolutions, to cause others to be elected; whereby the legislative
powers, incapable of annihilation, have returned to the people at large for their exercise; the state remaining
```

練習1-2 例外處理

1. 修改 ExceptionPractice 專案
2. 加入關閉 BufferedReader b 的程式碼
   - finally區段中加入
     - b.close();
   - 適當修改程式碼使其能編譯成功
3. 測試、執行
4. 改以try with resource 的方式建立可自動關閉的資源
5. 測試、執行

關閉資源

```java
ExceptionMain.java ✕

1  import java.io.*;
2  public class ExceptionMain {
3
4⊝     public static void main(String[] args){
5          try {
6              System.out.println("Reading from file:" + args[0]);
7          } catch(ArrayIndexOutOfBoundsException ex) {
8              System.err.println("未輸入讀取檔案名稱,程式結束");
9              System.exit(0);
10         }
11
12         BufferedReader b = null;
13         try {
14             b = new BufferedReader(new FileReader(args[0]));
15             String s = null;
16             while ((s = b.readLine()) != null) {
17                 System.out.println(s);
18             }
19         } catch(FileNotFoundException ex) {
20             System.err.println("檔案不存在:"+args[0]);
21         } catch(IOException ex) {
22             System.err.println("檔案讀取失敗:"+ex.getMessage());
23         } finally {
24             try {
25                 b.close();
26             } catch(IOException ex) {
27                 System.err.println("關閉檔案發生錯誤:"+ex.getMessage());
28             }
29         }
30     }
31
32 }
33
```

測試、執行

```
Reading from file:DeclarationOfIndependence.txt
When, in the course of human events, it becomes necessary for one people to dissolve the political bands
which have connected them with another, and to assume among the powers of the earth, the separate and equal
station to which the laws of nature and of nature's God entitle them, a decent respect to the opinions of
mankind requires that they should declare the causes which impel them to the separation.

We hold these truths to be self-evident, that all men are created equal, that they are endowed by their
Creator with certain unalienable rights, that among these are life, liberty and the pursuit of happiness.
That to secure these rights, governments are instituted among men, deriving their just powers from the consent
of the governed. That whenever any form of government becomes destructive to these ends, it is the right of
the people to alter or to abolish it, and to institute new government, laying its foundation on such principles
and organizing its powers in such form, as to them shall seem most likely to effect their safety and happiness.
Prudence, indeed, will dictate that governments long established should not be changed for light and transient
causes; and accordingly all experience hath shown that mankind are more disposed to suffer, while evils are
sufferable, than to right themselves by abolishing the forms to which they are accustomed. But when a long
train of abuses and usurpations, pursuing invariably the same object evinces a design to reduce them under
absolute despotism, it is their right, it is their duty, to throw off such government, and to provide new
guards for their future security. --Such has been the patient sufferance of these colonies; and such is now
the necessity which constrains them to alter their former systems of government. The history of the present
King of Great Britain is a history of repeated injuries and usurpations, all having in direct object the
establishment of an absolute tyranny over these states. To prove this, let facts be submitted to a candid
world.

He has refused his assent to laws, the most wholesome and necessary for the public good.

He has forbidden his governors to pass laws of immediate and pressing importance, unless suspended in their
operation till his assent should be obtained; and when so suspended, he has utterly neglected to attend to them.

He has refused to pass other laws for the accommodation of large districts of people, unless those people would
relinquish the right of representation in the legislature, a right inestimable to them and formidable to tyrants only.

He has called together legislative bodies at places unusual, uncomfortable, and distant from the depository of
their public records, for the sole purpose of fatiguing them into compliance with his measures.

He has dissolved representative houses repeatedly, for opposing with manly firmness his invasions on the rights
of the people.

He has refused for a long time, after such dissolutions, to cause others to be elected; whereby the legislative
powers, incapable of annihilation, have returned to the people at large for their exercise; the state remaining
```

資源自動關閉

```java
import java.io.*;
public class ExceptionMain {

    public static void main(String[] args){
        try {
            System.out.println("Reading from file:" + args[0]);
        } catch(ArrayIndexOutOfBoundsException ex) {
            System.err.println("未輸入讀取檔案名稱,程式結束");
            System.exit(0);
        }

        try (BufferedReader b = new BufferedReader(new FileReader(args[0]))){
            String s = null;
            while ((s = b.readLine()) != null) {
                System.out.println(s);
            }
        } catch(IOException ex) {
            System.err.println("檔案讀取失敗:"+ex.getMessage());
        }
    }

}
```

測試、執行

<terminated> ExceptionMain [Java Application] C:\Program Files\Java\jdk-17.0.4\bin\javaw.exe (2024年3月18日 下午9:14:11 – 下午9:14:11) [pid: 20792]

```
Reading from file:DeclarationOfIndependence.txt
When, in the course of human events, it becomes necessary for one people to dissolve the political bands
which have connected them with another, and to assume among the powers of the earth, the separate and equal
station to which the laws of nature and of nature's God entitle them, a decent respect to the opinions of
mankind requires that they should declare the causes which impel them to the separation.

We hold these truths to be self-evident, that all men are created equal, that they are endowed by their
Creator with certain unalienable rights, that among these are life, liberty and the pursuit of happiness.
That to secure these rights, governments are instituted among men, deriving their just powers from the consent
of the governed. That whenever any form of government becomes destructive to these ends, it is the right of
the people to alter or to abolish it, and to institute new government, laying its foundation on such principles
and organizing its powers in such form, as to them shall seem most likely to effect their safety and happiness.
Prudence, indeed, will dictate that governments long established should not be changed for light and transient
causes; and accordingly all experience hath shown that mankind are more disposed to suffer, while evils are
sufferable, than to right themselves by abolishing the forms to which they are accustomed. But when a long
train of abuses and usurpations, pursuing invariably the same object evinces a design to reduce them under
absolute despotism, it is their right, it is their duty, to throw off such government, and to provide new
guards for their future security. --Such has been the patient sufferance of these colonies; and such is now
the necessity which constrains them to alter their former systems of government. The history of the present
King of Great Britain is a history of repeated injuries and usurpations, all having in direct object the
establishment of an absolute tyranny over these states. To prove this, let facts be submitted to a candid
world.

He has refused his assent to laws, the most wholesome and necessary for the public good.

He has forbidden his governors to pass laws of immediate and pressing importance, unless suspended in their
operation till his assent should be obtained; and when so suspended, he has utterly neglected to attend to them.

He has refused to pass other laws for the accommodation of large districts of people, unless those people would
relinquish the right of representation in the legislature, a right inestimable to them and formidable to tyrants only.

He has called together legislative bodies at places unusual, uncomfortable, and distant from the depository of
their public records, for the sole purpose of fatiguing them into compliance with his measures.

He has dissolved representative houses repeatedly, for opposing with manly firmness his invasions on the rights
of the people.

He has refused for a long time, after such dissolutions, to cause others to be elected; whereby the legislative
powers, incapable of annihilation, have returned to the people at large for their exercise; the state remaining
```

練習2 自訂例外

1. 修改 EmployeeDAO 專案
2. 自訂com.example.dao.DAOException類別
   - 繼承Exception
   - 提供傳入字串message及包覆原例外的建構子
3. 修改EmployeeDAO介面
   - 實作AutoCloseable
   - 所有方法宣告丟出DAOException
4. 修改EmployeeDAOMemoryImpl類別
   - add(Employee emp)
     - 驗證 if(employeeArray[emp.getId()]!=null), 丟出DAOException
     - 使用 try-catch, 避免發生 ArrayIndexOutOfBoundsException
   - update(Employee emp)
     - 驗證 if(employeeArray[emp.getId()]==null), 丟出DAOException
     - 使用 try-catch, 避免發生 ArrayIndexOutOfBoundsException
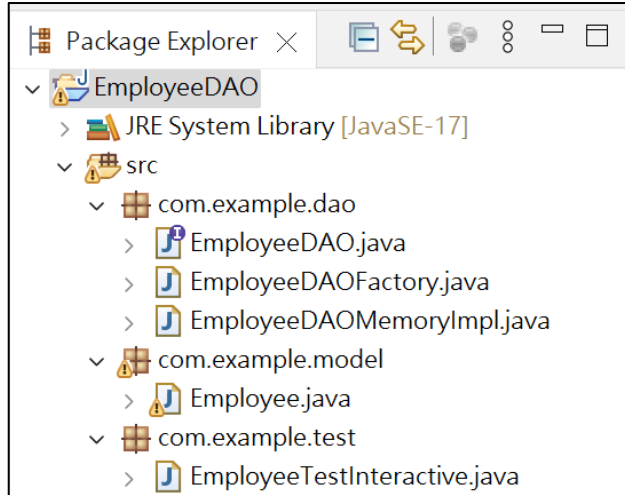
練習2 自訂例外

5) 修改EmployeeDAOMemoryImpl類別
- delete(int id)
  - 驗證 if(employeeArray[id]==null), 丟出DAOException
  - 使用 try-catch, 避免發生 ArrayIndexOutOfBoundsException
- findById(int id)
  - 使用 try-catch, 避免發生 ArrayIndexOutOfBoundsException
- 提供close() 不要丟出例外

6) 修改主類別 EmployeeTestInteractive
- 移除main()丟出的Exception
- executeMenu() 丟出 DAOException
- 使用try-with-resource包圍do-while loop
  - try-with-resource中建立EmployeeDAO及BufferedReader
  - 捕捉IOException, 發生例外程式結束
- try-catch包圍executeMenu(), 捕捉DAOException例外, 避免中斷
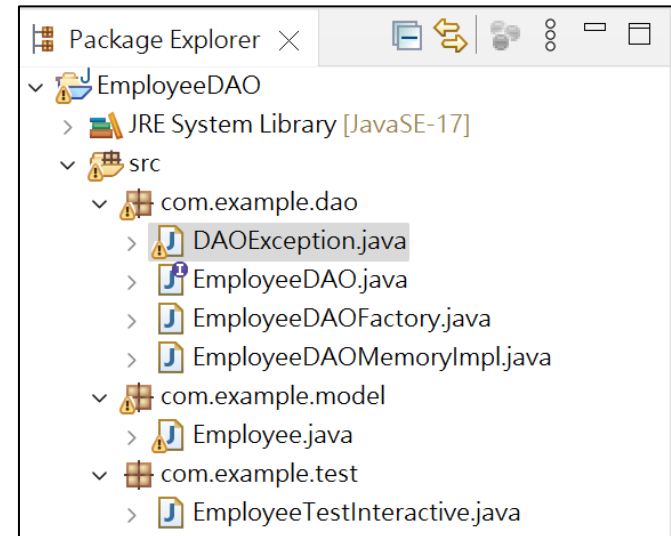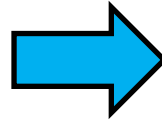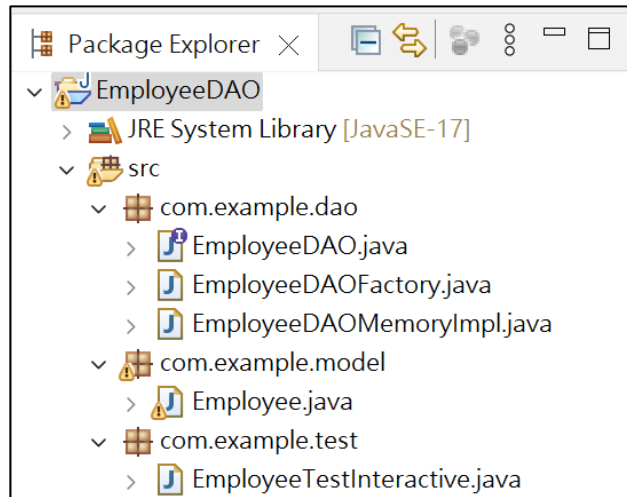
7) 測試、執行

EmployeeDAO 專案

測試、執行


```
Console  ✕

<terminated> EmployeeTestInteractive [Java Application] C:\Program Files\Java\jdk-17.0.4\bin\javaw.exe (2024年3月18日 下午9:32:57 – 下午9:33:21) [pid: 134


[C]reate | [R]ead | [U]pdate | [D]elete | [L]ist | [Q]uit:
C
Enter int value for employee id:
10
Enter value for employee first name :
Sean
Enter value for employee last name :
Cheng
Enter value for employee birth date (MMM d, yyyy) :
Mar 21, 1974
Enter float value for employee salary :
50000
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: Index 10 out of bounds for length 10
        at com.example.dao.EmployeeDAOMemoryImpl.add(EmployeeDAOMemoryImpl.java:13)
        at com.example.test.EmployeeTestInteractive.executeMenu(EmployeeTestInteractive.java:39)
        at com.example.test.EmployeeTestInteractive.main(EmployeeTestInteractive.java:19)
```

```
Console  ✕

<terminated> EmployeeTestInteractive [Java Application] C:\Program Files\Java\jdk-17.0.4\bin\javaw.exe (2024年3月18日 下午9:34:09 – 下午9:34:14) [pid: 154


[C]reate | [R]ead | [U]pdate | [D]elete | [L]ist | [Q]uit:
R
Enter int value for employee id:
-5
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: Index -5 out of bounds for length 10
        at com.example.dao.EmployeeDAOMemoryImpl.findById(EmployeeDAOMemoryImpl.java:28)
        at com.example.test.EmployeeTestInteractive.executeMenu(EmployeeTestInteractive.java:50)
        at com.example.test.EmployeeTestInteractive.main(EmployeeTestInteractive.java:19)
```
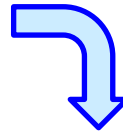
DAOException

com.example.dao.DAOException


```java
package com.example.dao;

public class DAOException extends Exception {

    public DAOException() {
        super();
    }

    public DAOException(String message) {
        super(message);
    }

    public DAOException(Throwable cause) {
        super(cause);
    }

    public DAOException(String message, Throwable cause) {
        super(message, cause);
    }

}
```

EmployeeDAO

```java
EmployeeDAO.java  ×

1  package com.example.dao;
2
3  import com.example.model.Employee;
4
5  public interface EmployeeDAO {
6
7      public void add(Employee emp);
8
9      public void update(Employee emp);
10
11     public void delete(int id);
12
13     public Employee findById(int id);
14
15     public Employee[] getAllEmployees();
16
17  }
18
```
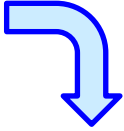
```java
EmployeeDAO.java  ×

1  package com.example.dao;
2
3  import com.example.model.Employee;
4
5  public interface EmployeeDAO extends AutoCloseable{
6
7      public void add(Employee emp) throws DAOException;
8
9      public void update(Employee emp) throws DAOException;
10
11     public void delete(int id) throws DAOException;
12
13     public Employee findById(int id) throws DAOException;
14
15     public Employee[] getAllEmployees() throws DAOException;
16
17  }
18
```
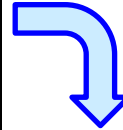
EmployeeDAOMemoryImpl

```java
EmployeeDAOMemoryImpl.java  ×

 1  package com.example.dao;
 2
 3⊕ import java.util.ArrayList;
 7
 8  public class EmployeeDAOMemoryImpl implements EmployeeDAO {
 9      private Employee[] employeeArray = new Employee[10];
10
11⊖     @Override
12      public void add(Employee emp) {
13          employeeArray[emp.getId()] = emp;
14      }
15
17⊕     public void update(Employee emp)
20
22⊕     public void delete(int id) {..
25
27⊕     public Employee findById(int id)
30
32⊕     public Employee[] getAllEmployee
42
43  }
44
```
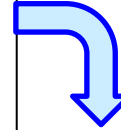
```java
EmployeeDAOMemoryImpl.java  ×

 1  package com.example.dao;
 2
 3⊖ import java.util.ArrayList;
 4  import java.util.List;
 5  import com.example.model.Employee;
 6
 7  public class EmployeeDAOMemoryImpl implements EmployeeDAO {
 8      private Employee[] employeeArray = new Employee[10];
 9
10⊖     @Override
11      public void add(Employee emp) throws DAOException{
12          int id = emp.getId();
13          try {
14              if(employeeArray[id]!=null)
15                  throw new DAOException("員工已存在,新增失敗!");
16              employeeArray[id] = emp;
17          } catch(ArrayIndexOutOfBoundsException ex) {
18              throw new DAOException("員工編號需小於10,新增失敗!", ex);
19          }
20      }
21
```

```
15
16⊖    @Override
17    public void update(Employee emp) {
18        employeeArray[emp.getId()] = emp;
19    }
20
21⊖    @Override
22    public void delete(int id) {
23        employeeArray[id] = null;
24    }
25
```

```
22⊖    @Override
23    public void update(Employee emp) throws DAOException{
24        int id = emp.getId();
25        try {
26            if(employeeArray[id]==null)
27                throw new DAOException("員工不存在,修改失敗!");
28            employeeArray[id] = emp;
29        } catch(ArrayIndexOutOfBoundsException ex) {
30            throw new DAOException("員工編號需小於10,修改失敗!", ex);
31        }
32    }
33
34⊖    @Override
35    public void delete(int id) throws DAOException{
36        try {
37            employeeArray[id] = null;
38        } catch(ArrayIndexOutOfBoundsException ex) {
39            throw new DAOException("員工編號需小於10,刪除失敗!", ex);
40        }
41    }
42
```

```java
26     @Override
27     public Employee findById(int id) {
28         return employeeArray[id];
29     }
30
31     @Override
32     public Employee[] getAllEmployees() {
33         List<Employee> emps = new ArrayList<>();
34         // Iterate through the memory array and find Employee objects
35         for (Employee e : employeeArray) {
36             if (e != null) {
37                 emps.add(e);
38             }
39         }
40         return emps.toArray(new Employee[0]);
41     }
42
43 }
44
```
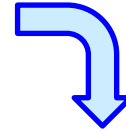
```java
43     @Override
44     public Employee findById(int id) throws DAOException{
45         try {
46             return employeeArray[id];
47         } catch(ArrayIndexOutOfBoundsException ex) {
48             throw new DAOException("員工編號需小於10,查詢失敗!", ex);
49         }
50     }
51
52     @Override
53     public Employee[] getAllEmployees() {
54         List<Employee> emps = new ArrayList<>();
55         // Iterate through the memory array and find Employee objects
56         for (Employee e : employeeArray) {
57             if (e != null) {
58                 emps.add(e);
59             }
60         }
61         return emps.toArray(new Employee[0]);
62     }
63
64     @Override
65     public void close() {
66         System.out.println("關閉資源......");
67     }
68
69 }
70
```

EmployeeTestInteractive

```java
EmployeeTestInteractive.java  ×

1  package com.example.test;
2
3  import java.io.*;
4  import com.example.model.*;
5  import java.util.*;
6  import java.text.*;
7  import com.example.dao.*;
8
9  public class EmployeeTestInteractive {
10
11     public static void main(String[] args) throws Exception {
12         EmployeeDAOFactory factory = new EmployeeDAOFactory();
13         EmployeeDAO dao = factory.createEmployeeDAO();
14
15         boolean timeToQuit = false;
16
17
18
```

```java
EmployeeTestInteractive.java  ×

 9  public class EmployeeTestInteractive {
10
11     public static void main(String[] args) {
12         EmployeeDAOFactory factory = new EmployeeDAOFactory();
13         EmployeeDAO dao = factory.createEmployeeDAO();
14
15         boolean timeToQuit = false;
16         try(BufferedReader in = new BufferedReader(new InputStreamReader(System.in))){
17             do {
18                 try {
19                     timeToQuit = executeMenu(in, dao);
20                 } catch(DAOException ex) {
21                     System.err.println("發生例外:"+ex.getMessage());
22                 }
23             } while (!timeToQuit);
24         } catch(IOException ex) {
25             System.err.println(ex.toString());
26             System.err.println("程式結束!");
27         }
28     }
29
30     public static boolean executeMenu(BufferedReader in, EmployeeDAO dao) throws IOException, DAOException {
```

測試、執行


```
Console  ×

EmployeeTestInteractive [Java Application] C:\Program Files\Java\jdk-17.0.4\bin\javaw.exe


[C]reate | [R]ead | [U]pdate | [D]elete | [L]ist | [Q]uit:
C
Enter int value for employee id:
10
Enter value for employee first name :
Sean
Enter value for employee last name :
Cheng
Enter value for employee birth date (MMM d, yyyy) :
Mar 21, 1974
Enter float value for employee salary :
50000
發生例外:員工編號需小於10,新增失敗!


[C]reate | [R]ead | [U]pdate | [D]elete | [L]ist | [Q]uit:
C
Enter int value for employee id:
1
Enter value for employee first name :
Sean
Enter value for employee last name :
Cheng
Enter value for employee birth date (MMM d, yyyy) :
Mar 21, 1974
Enter float value for employee salary :
50000
Successfully added Employee Record: 1


Created Employee ID:   1
Employee Name: Sean Cheng
Birth Date:    3月 21, 1974
Salary:        $50,000.00
```

```
[C]reate | [R]ead | [U]pdate | [D]elete | [L]ist | [Q]uit:
C
Enter int value for employee id:
1
Enter value for employee first name :
David
Enter value for employee last name :
Wang
Enter value for employee birth date (MMM d, yyyy) :
Dec 25, 1980
Enter float value for employee salary :
70000
發生例外:員工已存在,新增失敗!
```