Banking Management System



## A PROJECT REPORT

**Submitted By**

## Shivendra Singh Kushwah

**(1802914014)**

## Submitted in partial fulfillment of the Requirements for the Degree of

MASTER OF COMPUTER APPLICATIONS

## Under the Supervision of

**Submitted to**

**DEPARTMENT OF COMPUTER APPLICATIONS**

## KIET Group of Institutions, Ghaziabad Uttar Pradesh-201206

**(JUNE 2021)**

0

# DECLARATION

I hereby declare that the work presented in this report entitled “Banking Management System ", was carried out by me. I have not submitted the matter embodied in this report for the award of any other degree or diploma of another Institute or Institute.

I have given due credit to the original authors/sources for all the words, ideas, diagrams, graphics, computer programs, experiments, results, that are not my original contribution. I have used quotation marks to identify verbatim sentences and given credit to the original authors/sources.

I affirm that no portion of my work is plagiarized, and the experiments and results reported in the report are not manipulated. In the event of a complaint of plagiarism and the manipulation of the experiments and results, I shall be fully responsible and answerable.

Name : Shivendra Singh Kushwah

Enroll. No. : 1802914014

**(Shivendra Singh Kushwah)**

# CERTIFICATE

Certified that **Shivendra Singh Kushwah (1802914014)** has carried out the project work having “Banking Management System” for Master of Computer Applications from Dr. A.P.J. Abdul Kalam Technical University (AKTU**)** (formerly UPTU), Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student himself / herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other Institute/Institution.

**Dr. Sangeeta Arora External Examiner**

**Associate Professor**

**Dept. of Computer Applications**

**KIET Group of Institutions, Ghaziabad**

**Dr. Ajay Kumar Srivastava**

**Professor &amp; Head**

**Department of Computer Applications**

**KIET Group of Institutions, Ghaziabad**

**Date:**

# ABSTRACT

**“Banking Management System ”** is a JAVA application which is based on NET BEANS.

These days we are using databases in which coding is required to enter the data. But this time we have an application which can do our work more simpler and easier i.e JAVA. This application is used to store a huge amount of data properly and consistently.

The objective of this application is to show that how a normal person who doesn’t even know programming can use this application easily, it is flexible like data can be deleted enter or updated easily.A software project means a lot of experience. I learned a lot through this project. This project has sharpened our concept NET BEANS.

This concept of NET BEANS has now become a great role to play in today's technical world. These technologies will definitely take database systems far away .

Through this project I learnt so many things can be manage through this application like sales, marketing, commercials and many more things. The only drawback of JAVA is that it is expensive but to do great work we have to use good technology as today data security is the best and the most essential thing and JAVA contains that all. things which I also learnt make Data Flow Diagram, State Event Diagram and Activity Diagram with help of our project Mentors.

# ACKNOWLEDGEMENTS

Success in life is never attained single handedly. My deepest gratitude goes to my thesis supervisor, for her guidance, help and encouragement throughout my research work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to Dr. Ajay Kumar Shrivastava, Professor and Head, Department of Computer Applications, for his insightful comments and administrative help at various occasions.

Fortunately, I have many understanding friends, who have helped me a lot in many critical conditions.

Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

**Shivendra Singh Kushwah**

**Table of Content**

|  |  |  |
| --- | --- | --- |
| Certificate |  | i |
| Abstract |  | ii |
| Acknowledgements |  | iii |
| Table iv | of | Contents |

**CHAPTER 1: Introduction : 5-7**

1.1 Project details 1.2Purpose of the Project

**CHAPTER 2: Technical Feasibility : 14-23**

* 1. JAVA
  2. Standard Objects

**CAHPTER3: Creating Objects,Validations,Triggers : 23-28**

* 1. Creating JAVA Objects
  2. Creating a Validation Rule
  3. Form Triggers
     1. Form Trigger for single object
     2. Form Trigger for data transformation

**Output Screens : 29-34**

**References : 35**

## LIST OF FIGURES

**Fig No. Title Page No**

Fig 4.1 Screenshot of login page

Fig 4.2 Profile Form

Fig 4.2.1 Editing Profile Form

Fig 4.3 Deposit Form

Fig 4.4 Withdraw Form

Fig 4.5 View balance Form

Fig 4.6 Transfer Form

Fig 4.7 Customer list Form

Fig 4.8 Change pin Form

Fig 4.9 About Form

## CHAPTER 1

**INTRODUCTION**

## Project Details:

“**Banking Management System**” is taking a concept from “**NET BEANS**” and JAVA was founded by **“Marc Benioff”**.

As the JAVA is NET BEANS Application consists of a large, waist-level cabinet with number of clouds in it. Each cloud contains different objects through which we can access the data. Once the application start, we can enter the particular amount of data which we need to put in the application for sure, the individual can directly add the data to the particular objects and fields thereby adding the students data. This do not require much time only we have to filled and enter all the data related to particular student.

This application will be developed for PC’s , we can access it from anywhere and even from the phone easily using the JAVA application.

## Purpose:

JAVA provides you with the fastest path from Idea to App. You can concentrate on building your app using JAVA tools, rather than building the infrastructure and tools yourself. This can save you years of time and millions of dollars.

JAVA customers generally say that it’s unique for three major reasons:

**Fast** – Traditional CRM software can take more than a year to deploy, compare that to months or even weeks with JAVA.

**Easy** – JAVA wins in the easy to use category hands down. You can spend more time putting it to use and less time figuring it out.

**Effective** – Because it is easy to use and can be customized to meet business needs, customers find JAVA very effective.

JAVA is in the cloud, so your team can use it from anywhere with access to the internet. If you are a business that is rapidly changing or you are a seasoned company that’s been around for years, your business is probably changing too. JAVA is completely scalable to your growth.

JAVA seamlessly integrates with 3rd party apps. If you want to integrate JAVA with Gmail you can do it, if you want to integrate it with your accounting software you can do that too. On the other hand, integration is tough with other CRMs. JAVA is affordable, especially if you consider its vast variety of capabilities. Even startups and small businesses can use JAVA.

## Scope:

Students can use their device (laptop, mobile, desktop) or you can provide your lab.

We set up a branded solution for you (Software as a Service) and maintain every upgrade of the platform, right from hardware to software.

Our flagship product is built from scratch for engineering institutes with one objective – make students successful. Whether it is a campus job or higher education, we enable students and teachers to be successful by capturing, measuring, and analyzing the right data.

Engineering Students are at the core of the platform. We not only assess but also are part of the journey to accomplish the mission. Whether it is about finding a campus job or preparing for higher education, we enable you for success.

Students can learn from industry experts who are mentors on our platform (many of them are your college alumni). Take timely suggestions, join webinars and learn from their experiences. It is a virtual, ongoing, unstructured learning under supervised guidance that prepares you for the job ahead.

## CHAPTER 2

**NET BEANS**

## About:

NET BEANS relies on software for distributed batch and stream processing, as well as distributed storage. This chapter focuses on an oft*-*ignored angle of assuredness: performance assuredness. A significant pain point today is the inability to support reconfiguration operations, such as changing of the shard key in a sharded storage/database system, or scaling up (or down) of the number of virtual machines (VMs) being used in a stream or batch processing system. We discuss new techniques to support such reconfiguration operations in an online manner, whereby the system does not need to be shut down and the user/client*-*perceived behavior is indistinguishable regardless of whether a reconfiguration is occurring in the background, that is, the performance continues to be assured in spite of ongoing background reconfiguration. Next, we describe how to scale*-*out and scale*-*in (increase or decrease) the number of machines/VMs in NET BEANS frameworks like distributed stream processing and distributed graph processing systems, again while offering assured performance to the customer in spite of the reconfigurations occurring in the background. The ultimate performance assuredness is the ability to support SLAs/SLOs (service*-*level agreements/objectives) such as deadlines. We present a new real*-*time scheduler that supports priorities and hard deadlines for Hadoop jobs. We implemented our reconfiguration systems as patches to several popular and open*-*source NET BEANS systems, including MongoDB and Cassandra (storage), Storm (stream processing), LFGraph (graph processing), and Hadoop (batch processing).[1]

NET BEANS has become the industry standard for rapid application deployment, scalable server support, mobile and distributed services, and it provides access to (theoretically) infinite resources. Unfortunately, researchers are still trying to converge towards cross-provider NET BEANS frameworks to enable compatibility and seamless resource transition between cloud providers. Moreover, users are restricted to using the provider-specific pre-configured options of resources and services, irrespective of their current needs. At the same time, cloud services are provided as a direct service from the providers to the clients. This creates a segregated cloud market clientele, and non-negotiable pricing

strategies for the cloud services. In this paper, we propose Jugo, a generic architecture for cloud composition and negotiated service delivery for cloud users. Jugo acts as a match-maker for service specifications from the users with the currently available assets from the cloud providers. The engagement of a middle-man as an opaque cloud service provider will create a better opportunity for cloud users to find cheaper deals, price-matching, and flexible resource specifications, with increased revenue and higher resource utilization for the cloud service providers.[2]

Many enterprises in industries start using NET BEANS for their IT infrastructure services. This adoption of NET BEANS is a part of the enterprise transformation which is the migration from a legacy IT environment to NET BEANS. On the other hand, one of major targets is an industry solution which provides a critical business service to their end customers. This paper proposes Industry Cloud which is the enhanced design of NET BEANS for industry solutions. It efficiently supports industry solutions for enterprise business requirements. The paper describes Industry Cloud with a requirement analysis of industry solutions, those adopted functions, and three use case scenarios in the electronics and retail industry. The contribution of the paper is the analysis of industry wide requirements, the definition of Industry Cloud with a common function among industry solutions and the usage with use case scenarios.[3]

The complexity of Cloud infrastructures is increasing every year, requiring new concepts and tools to face off topics such as process configuration and reconfiguration, automatic scaling, elastic computing and healthiness control. This paper presents a Smart Cloud solution based on a Knowledge Base, KB, with the aim of modeling cloud resources, Service Level Agreements and their evolution, while enabling the reasoning on cloud structures and implementing strategies of efficient smart cloud management and intelligence. The solution proposed is composed of Smart Cloud Engine, SCE, the Knowledge Base, KB, and the Supervisor and Monitoring module for data acquisition. It can be easily integrated with any cloud configuration manager, cloud orchestra or, and monitoring tool, since the connections with these tools are performed by using REST calls and XML files.[4]

Current NET BEANS is primarily based on proprietary data centers, where hundreds of thousands of dedicated servers are setup to host the cloud services. In addition to the huge number of dedicated servers deployed in data centers, there are billions of underutilized Personal Computers (PCs), usually used only for a few hours per day, owned by individuals and organizations worldwide. The vast untapped compute and storage capacities of the underutilized PCs can be consolidated as alternative cloud fabrics to provision broad cloud services, primarily infrastructure as a service. This approach, thus referred to as "no data center" approach, complements the data center based cloud provision model. In this paper, we present our opportunistic NET BEANS system, called cuCloud, that runs on scavenged resources of underutilized PCs within an organization/community. Our system demonstrates that the "no data center" solution indeed works. Besides proving our concept, model, and philosophy, our experimental results are highly encouraging.[5]

Whatever one public cloud, private cloud or a mixed cloud, the users lack of effective security quantifiable evaluation methods to grasp the security situation of its own information infrastructure on the whole. This paper provides a quantifiable security evaluation system for different clouds that can be accessed by consistent API. The evaluation system includes security scanning engine, security recovery engine, security quantifiable evaluation model, visual display module and etc. The security evaluation model composes of a set of evaluation elements corresponding different fields, such as computing, storage, network, maintenance, application security and etc. Each element is assigned a three tuple on vulnerabilities, score and repair method. The system adopts “One vote vetoed” mechanism for one field to count its score and adds up the summary as the total score, and to create one security view.

We implement the quantifiable evaluation for different cloud users based on our G-Cloud platform. It shows the dynamic security scanning score for one or multiple clouds with visual graphs and guided users to modify configuration, improve operation and repair vulnerabilities, so as to improve the security of their cloud resources.[6]

To move applications to the cloud is not only a technical decision but also a business-oriented decision, in which both business and technical factors (e.g. transformation effort,

multi-tenancy and auto-scaling enablement, scalability and extensibility) should be considered. However, existing approaches and tools do not support a consumable business oriented cloud transformation decision to select more suitable transformation solution with the right cloud delivery model, services type, affordable transformation effort and etc. In this paper, we introduce a practical three-step approach and a tool, CTA (Cloud Transformation Advisor) to enable decision makers to identify the most suitable cloud transformation solution to satisfy their business goals based on a well-structured cloud transformation knowledge base.[7]

**CHAPTER 2**

**JAVA**

The JAVA Platform stores data in relational tables. The records in these tables contain data for the structure of the platform itself as well as user created data. For example, the data about the configuration and settings of an account are already in-built as a relational table. But you can also create your own tables to store data specific to your business like the 'dispatch schedule' for a week assuming you are a courier company.

These relational tables are roughly referred to as API Objects or only objects in JAVA.

There are three kinds of JAVA objects.

* Standard Objects − The objects already created for you by the JAVA platform.
* Custom Objects − These are the objects created by you based on your business processes.
* External Objects − The objects which you create map to the data stored outside your organization.
  1. **Standard Objects**

These are the objects which already exist in the JAVA platform to manage the configurations and settings of the environment. Once you log in to the JAVA platform, you can see the available objects.

**Important Standard Objects**

In this section, we will discuss the important standard objects in JAVA. The following table lists down the objects −

|  |  |  |
| --- | --- | --- |
| **bject Name** | **Meaning** | **Usage** |
| Account | Represents an individual account, which is an organization or person involved in the business like customers,  competitors,  partners, etc. | Use this object to query and manage accounts in your organization. |
| Account History | Represents the history of changes to the values in the fields of an account. | Use this object to identify changesto an  account. |
| Case | Represents a case, which is a customer issue or problem. | Use the case object to manage casesfor your organization. |
| Contact | Represents a contact, which is an individual associated with anaccount. | This object is used to manage individuals who are associated with an Account in the organization. |

|  |  |  |
| --- | --- | --- |
| ser | Represents a user in the organization. | This object is used to query information about users and also helps to provide and modify the information concerning the users. |
| Asset | Represents an item of commercial value, such as a product sold by the company or a competitor that a customer has purchased and installed. | This object is used to track assets previously sold into customer accounts. With asset tracking, a client application can quickly determine which products were previously sold or are currently installed at a specific account. |
| Domain | Read-only object that represents a custom Web address assigned to a site in your organization. | This read-only object is used to object to query the domains that are associated with each website in your organization. |

As IT technology advanced, a new style of innovation emerged, in which a leading innovation company invites end-users to its open software service platform. With respect to this type of innovation, a lot of innovation studies were performed to understand the structure of the interaction among users and the platform provider from the perspective of network science. By concentrating only on the internal mechanisms among agents, the previous studies miss to consider innovation through collective intelligence. A platform provider plays an important role in the innovation. In this research, we investigate the structure of a service network with empirical data gathered from JAVA.com App Exchange and discuss the role of a platform provider in innovation through collective intelligence. Our results suggest that the platform provider led the innovation in the initial period and, then, third party developers became gradually innovation leaders. Our findings are expected to re-orient the research focus from internal mechanisms to the role of platform providers.[8]

With the appearance of distributed computing, associations are hoping to move their Customer Relationship Management (CRM) applications from an On-Premise environment or we can say local servers to an On-Demand environment that is on cloud server. On-Premise environment is when association has the framework and programming inside their system. In On-Demand environment, an outsider has the base and programming and charges the relationship in light of its participation. JAVA is the principle On-Demand CRM thing.[9]

The advantages of cloud while supporting real-time service systems using the JAVA platform. We build here a service management platform for the Polish Billiards and Snooker Association (PBSA), based on a real-time system located in a cloud. It allows PBSA managers to accomplish tasks in this system on-demand. And, it is deployed as a private cloud to grant an access only to the employees from the snooker organization.[10]

In a recent scenario, IT industries are growing with the help of proper Utilization of available resources. The IT giants like Microsoft, Infosys, IBM, Oracle, & TCS are switching from theirs on premises IT setups to the cloud. NET BEANS is replacing the traditional model in which software applications installed on on-premise hardware, from desktop computers to server rooms, depending on the size of the business. The proposed work is about the cloud platform which is going to change all the traditional views of software, application, and product development Technologies. JAVA.com is one of the best cloud providers available in the recent scenario. There are number of reasons why IT industries are switching to the Cloud. And there are number of reasons why Industries have to think to adopt JAVA.com cloud. The proposed work is about to focus on important and common features of salsforce.com. These features are common for any developer to learn and use in to software, application and product development in JAVA.com. The goal of this proposed work is to show the available resources in the JAVA.com which are still new for the developers. This an approach to make people familiar with the JAVA.com cloud provider.[11]

Summary form only given as follows. The strongest predictors of new product success is understanding market requirements early in the new product development (NPD) process. A direct JAVA

is one of the best sources of new product ideas and market information, although not fully leveraged in many firms. A recent study of 248 salespeople in nineteen high-tech firms indicates wide variation in NPD involvement across and within a firm's JAVA. The study revealed that at the organization level, the length of the NPD cycle was associated with JAVA-initiated and NPD-headquarters initiated activities. In particular, the longer a firm's NPD cycle for product improvements, the lower the involvement the JAVA has in headquarters initiated involvement activities. A significant relationship exists between a firm's new product cycle time and the level of involvement in JAVA-initiated NPD predevelopment activities. At the salesperson level, several factors affect involvement in NPD activities. Most significant was the relationship between salesperson involvement and the distance between a salesperson's location and the NPD office site. The study shows that organizations can affect the degree of involvement that their JAVA or individual salespeople have in early phases of NPD.[12]

Motivation is important because it determines and individual's effort toward performing a task and it leads to an optimistic and challenging attitude at work place. Incentive motivation is concerned with the way goals influence behavior. Incentive is the promise or stimulus for greater action. Incentives are something that is given in addition to wages. It means additional remuneration or benefit to an employee in recognition of achievement or better work. Organizations in emerging markets are more likely to give group rewards and generally do not encourage risk taking as much as developed economies. Money motivates employees to some extent but there are other powerful sources of motivation such as interesting and challenging tasks and assignments, praise, good interpersonal relations etc. This paper analyzes incentives plan structure of sales people in Croatia. Advantages and disadvantages of different compensation plans and benefits for salespeople are discussed.[13]

Modern business decision models are often very complicated due to a deluge of information. Evaluation and diagnostics of such decision models is extremely challenging due to many factors, including the complexity and volume of data. In addition, since there is no ideal data sample to construct a control group for comparison studies, performance evaluation and diagnostics of business actions can easily be distorted by selection bias. In this paper, we design a framework to analyze this sample bias issue under a practical business scenario. In particular, we focus on: a) identification of the key factors which drive selection bias during the business decision; b) evaluation of the performance of business actions with consideration of the identified selection bias. We evaluate baseline analytics tools on the worldwide sales- force data of a large global corporation and clearly demonstrate that the selection bias issue makes the

usual evaluation very unstable and not trustable. However, by removing such detected sample bias, our framework can generate reasonable diagnostics results across different dimensions. The implemented analysis tool was applied to a worldwide business opportunity dataset of a multinational Fortune 500 corporation; the analytics results clearly show the significance of such a bias detection-based evaluation framework for sales-force optimization.[14]

Estimating generalizable relationships between actions and results from historical samples, especially when there is a level of noise or randomness in that signal, is an important problem to address before making decisions on actions to take. Many business analytics problems require the optimal assignment of limited resources to actions and activities to maximize some result or objective such as profit. We present a novel approach to solving this class of analytics problems by modeling the relationship between resource effort and expected return as a dose-response signal and formulating its causal estimation as one of kernel regression. The estimated expected value and variance of the result are then used to optimize resource allocation so as to maximize expected response while minimizing the risk around response subject to business constraints. We apply this approach to the task of optimally assigning salespeople to enterprise clients using real-world data, and show that profit can be substantially increased with fewer salespeople and reduced risk.[15]

**CHAPTER 3**

**HOW TO CREATE OBJECTS,VALIDATIONS AND TRIGGERS**

* 1. **Creating a JAVA Custom Object**
* First, follow this path: Setup > Build > Create > Objects > on the screen, click on the new custom object button > enter label name, plural label, and object name > enter record name as a data type.
* To create a record name, first, consider the two data types:
* Text
* Auto-number

You also have some optional features while creating objects:

* Allow reports: If you tick off this checkbox, then only these objects will be available to make reports.
* Allow activities: If you tick off this checkbox, then you are able to make activities on this object.
* Track field history: When you tick off this checkbox, then you can merely track fields. You can track up to 20 fields for a single object.

Now, you have the following deployment modes:

* In development: If you opt for this, then this object will remain in the development mode. It will not be present for deployment.
* Deployed: When you opt for this, the object will be available for deployment.

The following options are available only when you are creating a custom object for the first time:

* Add notes and attachments related list
* Launch a new custom tab wizard after saving this custom object After completing all these settings, press the Save button.

If you do not select ‘Launch new custom tab wizard’ from the object creation page, then the object would be saved without the tab appearance. In such a case, first, you need to make a tab for this object. On the

other hand, if you select this option, the object would be saved, along with a tab would be created, which will be visible to you.

* 1. **Creating a Validation Rule**
* From Setup, go to Object Manager and click Account
* In the left sidebar, click Validation Rules.
* Click New.
* Enter the following properties for your validation rule
* Rule Name: Account\_Number\_8\_Characters
* Error Condition Formula:
* LEN( AccountNumber) != 8
* Error Message: Account number must be 8 characters long.
* To check your formula for errors, click Check Syntax.
* Click Save to finish.
  1. **Form Triggers**

The following steps show the different types of triggers you can use.

* From Setup, select Customize and then click the object that you want to add the trigger to.
* Click Triggers and then click New.
* To define your trigger, enter Form code similar to this sample code.
  + Replace [ObjectName] with the name of the object that you are adding the trigger to.
  + Replace [NameOfCustomSetting] with the integration name you created on the Einstein Discovery Integration information page.
    1. **Form coding of Account\_test.java**

**import static java.lang.Thread.sleep;**

**import javax.swing.JTextField;**

**import org.junit.After;**

**import org.junit.AfterClass;**

**import org.junit.Before;**

**import org.junit.BeforeClass;**

**import org.junit.Test;**

**import static org.junit.Assert.\*;**

**public class AccountTest {**

**private JTextField inputTest;**

**public AccountTest() {**

**}**

**@BeforeClass**

**public static void setUpClass() {**

**}**

**@AfterClass**

**public static void tearDownClass() {**

**}**

**@Before**

**public void setUp() {**

**}**

**@After**

**public void tearDown() {**

**}**

**@Test**

**public void testRandomAcc() {**

**System.out.println("randomAcc");**

**Account instance = new Account();**

**instance.randomAcc();**

**}**

**@Test**

**public void testRandomMICR() {**

**System.out.println("randomMICR");**

**Account instance = new Account();**

**instance.randomMICR();**

**// TODO review the generated test code and remove the default call to fail.**

**}**

**@Test**

**public void testRandomPin() {**

**System.out.println("randomPin");**

**Account instance = new Account();**

**instance.randomPin();**

**}**

**@Test**

**public void testBal() {**

**System.out.println("Bal");**

**Account instance = new Account();**

**instance.Bal();**

**}**

**@Test**

**public void testMain() {**

**System.out.println("main");**

**String[] args = null;**

**Account.main(args);**

**}**

**}**

}

* + 1. **Form coding of Mypage\_test.java**

**import org.junit.After;**

**import org.junit.AfterClass;**

**import org.junit.Before;**

**import org.junit.BeforeClass;**

**import org.junit.Test;**

**import static org.junit.Assert.\*;**

**public class MyPageTest {**

**public MyPageTest() {**

**}**

**@BeforeClass**

**public static void setUpClass() {**

**}**

**@AfterClass**

**public static void tearDownClass() {**

**}**

**@Before**

**public void setUp() {**

**}**

**@After**

**public void tearDown() {**

**}**

**@Test**

**public void testCalendar() {**

**System.out.println("Calendar");**

**MyPage instance = new MyPage();**

**instance.Calendar();**

**}**

**@Test**

**public void testTable1() {**

**System.out.println("Table1");**

**MyPage instance = new MyPage();**

**instance.Table1();**

**}**

**@Test**

**public void testAccount() {**

**System.out.println("Account");**

**MyPage instance = new MyPage();**

**instance.Account();**

**}**

**@Test**

**public void testTransferC() {**

**System.out.println("TransferC");**

**MyPage instance = new MyPage();**

**instance.TransferC();**

**}**

**@Test**

**public void testTransferD() {**

**System.out.println("TransferD");**

**MyPage instance = new MyPage();**

**instance.TransferD();**

**}**

**@Test**

**public void testMain() {**

**System.out.println("main");**

**String[] args = null;**

**MyPage.main(args);**

**}**

**}**

* + 1. **Form coding of Account.java**

**/\***

**\* To change this license header, choose License Headers in Project Properties.**

**\* To change this template file, choose Tools | Templates**

**\* and open the template in the editor.**

**\*/**

**import java.sql.Connection;**

**import org.junit.After;**

**import org.junit.AfterClass;**

**import org.junit.Before;**

**import org.junit.BeforeClass;**

**import org.junit.Test;**

**import static org.junit.Assert.\*;**

**/\*\***

**\***

**\* @author aqua\_**

**\*/**

**public class dbconnectTest {**

**public dbconnectTest() {**

**}**

**@BeforeClass**

**public static void setUpClass() {**

**}**

**@AfterClass**

**public static void tearDownClass() {**

**}**

**@Before**

**public void setUp() {**

**}**

**@After**

**public void tearDown() {**

**}**

**/\*\***

**\* Test of connectDb method, of class dbconnect.**

**\*/**

**@Test**

**public void testConnectDb() {**

**System.out.println("connectDb");**

**Connection expResult = null;**

**Connection result = dbconnect.connectDb();**

**assertEquals(expResult, result);**

**// TODO review the generated test code and remove the default call to fail.**

**}**

**}**

* + 1. **Form coding of Loading.java**

**import java.sql.Connection;**

**import java.sql.PreparedStatement;**

**import java.sql.ResultSet;**

**import javax.swing.\*;**

**/\***

**\* Document: Loading.java**

**\* Unitec Institute of Technology**

**\* @authors Natasha Bettridge, Taylor Tran and Michael Yin**

**\* Document: Contains the Create account form information from pt 4**

**\*/**

**/\*\***

**\***

**\* @author Natasha Bettridge, Taylor Tran, Michael Yin**

**\*/**

**public class Loading extends javax.swing.JFrame implements Runnable {**

**Connection conn;**

**ResultSet rs;**

**PreparedStatement pst;**

**int s =0;**

**Thread th;**

**/\*\***

**\* Creates new form Loading**

**\*/**

**public Loading() {**

**super("Loading");**

**initComponents();**

**th=new Thread((Runnable)this);**

**}**

**public void setUpLoading(){**

**setVisible(false);**

**th.start();**

**}**

**public void run(){**

**try{**

**for(int i=0;i<=200;i++){**

**s++;**

**int m = Progressbar.getMaximum();**

**int v = Progressbar.getValue();**

**if(v<m){**

**Progressbar.setValue(Progressbar.getValue()+1);**

**}else{**

**i = 201;**

**setVisible(false);**

**MyPage ob = new MyPage();**

**ob.setVisible(true);**

**}**

**Thread.sleep(50);**

**}**

**}catch(Exception e){**

**}**

**}**

**/\*\***

**\* This method is called from within the constructor to initialize the form.**

**\* WARNING: Do NOT modify this code. The content of this method is always**

**\* regenerated by the Form Editor.**

**\*/**

**@SuppressWarnings("unchecked")**

**// <editor-fold defaultstate="collapsed" desc="Generated Code">**

**private void initComponents() {**

**LoadingPage = new javax.swing.JPanel();**

**Loadingtxt = new javax.swing.JLabel();**

**Progressbar = new javax.swing.JProgressBar();**

**Loading = new javax.swing.JLabel();**

**jLabel1 = new javax.swing.JLabel();**

**setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);**

**LoadingPage.setBackground(new java.awt.Color(255, 255, 255));**

**LoadingPage.setBorder(new javax.swing.border.LineBorder(new java.awt.Color(21, 21, 48), 3, true));**

**Loadingtxt.setFont(new java.awt.Font("Tahoma", 1, 12)); // NOI18N**

**Loadingtxt.setText("Loading....");**

**Progressbar.setForeground(new java.awt.Color(36, 51, 97));**

**Progressbar.setStringPainted(true);**

**Loading.setIcon(new javax.swing.ImageIcon(getClass().getResource("/BankingImages/icon-swirl-big.png"))); // NOI18N**

**jLabel1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/xxx.png"))); // NOI18N**

**javax.swing.GroupLayout LoadingPageLayout = new javax.swing.GroupLayout(LoadingPage);**

**LoadingPage.setLayout(LoadingPageLayout);**

**LoadingPageLayout.setHorizontalGroup(**

**LoadingPageLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)**

**.addGroup(LoadingPageLayout.createSequentialGroup()**

**.addGroup(LoadingPageLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)**

**.addComponent(jLabel1)**

**.addGroup(LoadingPageLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)**

**.addGroup(LoadingPageLayout.createSequentialGroup()**

**.addGap(211, 211, 211)**

**.addComponent(Loadingtxt))**

**.addGroup(LoadingPageLayout.createSequentialGroup()**

**.addGap(81, 81, 81)**

**.addComponent(Progressbar, javax.swing.GroupLayout.PREFERRED\_SIZE, 329, javax.swing.GroupLayout.PREFERRED\_SIZE))**

**.addGroup(LoadingPageLayout.createSequentialGroup()**

**.addGap(192, 192, 192)**

**.addComponent(Loading, javax.swing.GroupLayout.PREFERRED\_SIZE, 105, javax.swing.GroupLayout.PREFERRED\_SIZE))))**

**.addContainerGap(81, Short.MAX\_VALUE))**

**);**

**LoadingPageLayout.setVerticalGroup(**

**LoadingPageLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)**

**.addGroup(LoadingPageLayout.createSequentialGroup()**

**.addGap(61, 61, 61)**

**.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 119, javax.swing.GroupLayout.PREFERRED\_SIZE)**

**.addGap(49, 49, 49)**

**.addComponent(Progressbar, javax.swing.GroupLayout.PREFERRED\_SIZE, 39, javax.swing.GroupLayout.PREFERRED\_SIZE)**

**.addGap(18, 18, 18)**

**.addComponent(Loadingtxt)**

**.addGap(32, 32, 32)**

**.addComponent(Loading, javax.swing.GroupLayout.PREFERRED\_SIZE, 79, javax.swing.GroupLayout.PREFERRED\_SIZE)**

**.addContainerGap(186, Short.MAX\_VALUE))**

**);**

**javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());**

**getContentPane().setLayout(layout);**

**layout.setHorizontalGroup(**

**layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)**

**.addComponent(LoadingPage, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)**

**);**

**layout.setVerticalGroup(**

**layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)**

**.addComponent(LoadingPage, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)**

**);**

**setSize(new java.awt.Dimension(513, 643));**

**setLocationRelativeTo(null);**

**}// </editor-fold>**

**/\*\***

**\* @param args the command line arguments**

**\*/**

**public static void main(String args[]) {**

**/\* Set the Nimbus look and feel \*/**

**//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">**

**/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.**

**\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html**

**\*/**

**try {**

**for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {**

**if ("Nimbus".equals(info.getName())) {**

**javax.swing.UIManager.setLookAndFeel(info.getClassName());**

**break;**

**}**

**}**

**} catch (ClassNotFoundException ex) {**

**java.util.logging.Logger.getLogger(Loading.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);**

**} catch (InstantiationException ex) {**

**java.util.logging.Logger.getLogger(Loading.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);**

**} catch (IllegalAccessException ex) {**

**java.util.logging.Logger.getLogger(Loading.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);**

**} catch (javax.swing.UnsupportedLookAndFeelException ex) {**

**java.util.logging.Logger.getLogger(Loading.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);**

**}**

**//</editor-fold>**

**/\* Create and display the form \*/**

**java.awt.EventQueue.invokeLater(new Runnable() {**

**public void run() {**

**new Loading().setVisible(true);**

**}**

**});**

**}**

**// Variables declaration - do not modify**

**private javax.swing.JLabel Loading;**

**private javax.swing.JPanel LoadingPage;**

**private javax.swing.JLabel Loadingtxt;**

**private javax.swing.JProgressBar Progressbar;**

**private javax.swing.JLabel jLabel1;**

**// End of variables declaration**

**}**

* + 1. **Form coding of Authentication.java**

**import org.junit.After;**

**import org.junit.AfterClass;**

**import org.junit.Before;**

**import org.junit.BeforeClass;**

**import org.junit.Test;**

**import static org.junit.Assert.\*;**

**/\*\***

**\***

**\* @author aqua\_**

**\*/**

**public class AuthenticationTest {**

**public AuthenticationTest() {**

**}**

**@BeforeClass**

**public static void setUpClass() {**

**}**

**@AfterClass**

**public static void tearDownClass() {**

**}**

**@Before**

**public void setUp() {**

**}**

**@After**

**public void tearDown() {**

**}**

**/\*\***

**\* Test of main method, of class Authentication.**

**\*/**

**@Test**

**public void testMain() {**

**System.out.println("main");**

**String[] args = null;**

**Authentication.main(args);**

**// TODO review the generated test code and remove the default call to fail.**

**}**

**}**

* + 1. **Form coding of Loading test.java**

**/\***

**\* To change this license header, choose License Headers in Project Properties.**

**\* To change this template file, choose Tools | Templates**

**\* and open the template in the editor.**

**\*/**

**import org.junit.After;**

**import org.junit.AfterClass;**

**import org.junit.Before;**

**import org.junit.BeforeClass;**

**import org.junit.Test;**

**import static org.junit.Assert.\*;**

**/\*\***

**\***

**\* @author aqua\_**

**\*/**

**public class LoadingTest {**

**public LoadingTest() {**

**}**

**@BeforeClass**

**public static void setUpClass() {**

**}**

**@AfterClass**

**public static void tearDownClass() {**

**}**

**@Before**

**public void setUp() {**

**}**

**@After**

**public void tearDown() {**

**}**

**@Test**

**public void testSetUpLoading() {**

**System.out.println("setUpLoading");**

**Loading instance = new Loading();**

**instance.setUpLoading();**

**// TODO review the generated test code and remove the default call to fail.**

**}**

**@Test**

**public void testRun() {**

**System.out.println("run");**

**Loading instance = new Loading();**

**instance.run();**

**}**

**@Test**

**public void testMain() {**

**System.out.println("main");**

**String[] args = null;**

**Loading.main(args);**

**}**

**}**

* + 1. **Form coding of mypagetest.java**

**/\***

**\* To change this license header, choose License Headers in Project Properties.**

**\* To change this template file, choose Tools | Templates**

**\* and open the template in the editor.**

**\*/**

**import org.junit.After;**

**import org.junit.AfterClass;**

**import org.junit.Before;**

**import org.junit.BeforeClass;**

**import org.junit.Test;**

**import static org.junit.Assert.\*;**

**/\*\***

**\***

**\* @author aqua\_**

**\*/**

**public class MyPageTest {**

**public MyPageTest() {**

**}**

**@BeforeClass**

**public static void setUpClass() {**

**}**

**@AfterClass**

**public static void tearDownClass() {**

**}**

**@Before**

**public void setUp() {**

**}**

**@After**

**public void tearDown() {**

**}**

**/\*\***

**\* Test of Calendar method, of class MyPage.**

**\***

**@Test**

**public void testCalendar() {**

**System.out.println("Calendar");**

**MyPage instance = new MyPage();**

**instance.Calendar();**

**// TODO review the generated test code and remove the default call to fail.**

**}**

**/\*\***

**\* Test of Table1 method, of class MyPage.**

**\***

**@Test**

**public void testTable1() {**

**System.out.println("Table1");**

**MyPage instance = new MyPage();**

**instance.Table1();**

**// TODO review the generated test code and remove the default call to fail.**

**}**

**/\*\***

**\* Test of Account method, of class MyPage.**

**\***

**@Test**

**public void testAccount() {**

**System.out.println("Account");**

**MyPage instance = new MyPage();**

**instance.Account();**

**// TODO review the generated test code and remove the default call to fail.**

**}**

**/\*\***

**\* Test of TransferC method, of class MyPage.**

**\***

**@Test**

**public void testTransferC() {**

**System.out.println("TransferC");**

**MyPage instance = new MyPage();**

**instance.TransferC();**

**// TODO review the generated test code and remove the default call to fail.**

**}**

**/\*\***

**\* Test of TransferD method, of class MyPage.**

**\***

**@Test**

**public void testTransferD() {**

**System.out.println("TransferD");**

**MyPage instance = new MyPage();**

**instance.TransferD();**

**// TODO review the generated test code and remove the default call to fail.**

**}**

**/\*\***

**\* Test of main method, of class MyPage.**

**\*/**

**@Test**

**public void testMain() {**

**System.out.println("main");**

**String[] args = null;**

**MyPage.main(args);**

**// TODO review the generated test code and remove the default call to fail.**

**}**

**}**

* + 1. **Form coding of DBconnect test.java**

**/\***

**\* To change this license header, choose License Headers in Project Properties.**

**\* To change this template file, choose Tools | Templates**

**\* and open the template in the editor.**

**\*/**

**import java.sql.Connection;**

**import org.junit.After;**

**import org.junit.AfterClass;**

**import org.junit.Before;**

**import org.junit.BeforeClass;**

**import org.junit.Test;**

**import static org.junit.Assert.\*;**

**/\*\***

**\***

**\* @author aqua\_**

**\*/**

**public class dbconnectTest {**

**public dbconnectTest() {**

**}**

**@BeforeClass**

**public static void setUpClass() {**

**}**

**@AfterClass**

**public static void tearDownClass() {**

**}**

**@Before**

**public void setUp() {**

**}**

**@After**

**public void tearDown() {**

**}**

**/\*\***

**\* Test of connectDb method, of class dbconnect.**

**\*/**

**@Test**

**public void testConnectDb() {**

**System.out.println("connectDb");**

**Connection expResult = null;**

**Connection result = dbconnect.connectDb();**

**assertEquals(expResult, result);**

**// TODO review the generated test code and remove the default call to fail.**

**}**

**}**

* + 1. **Form coding of UImanager.java**

**/\***

**\* To change this license header, choose License Headers in Project Properties.**

**\* To change this template file, choose Tools | Templates**

**\* and open the template in the editor.**

**\*/**

**/\*\***

**\***

**\* @author Natas**

**\*/**

**class UIManager {**

**static void setLookAndFeel(String comjtattooplafsmartSmartLookAndFeel) {**

**throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.**

**}**

**}**

* + 1. **Form coding of Newjframe.java**

**/\***

**\* To change this license header, choose License Headers in Project Properties.**

**\* To change this template file, choose Tools | Templates**

**\* and open the template in the editor.**

**\*/**

**/\*\***

**\***

**\* @author Shivendra Singh**

**\*/**

**public class NewJFrame extends javax.swing.JFrame {**

**/\*\***

**\* Creates new form NewJFrame**

**\*/**

**public NewJFrame() {**

**initComponents();**

**}**

**/\*\***

**\* This method is called from within the constructor to initialize the form.**

**\* WARNING: Do NOT modify this code. The content of this method is always**

**\* regenerated by the Form Editor.**

**\*/**

**@SuppressWarnings("unchecked")**

**// <editor-fold defaultstate="collapsed" desc="Generated Code">**

**private void initComponents() {**

**jLabel1 = new javax.swing.JLabel();**

**setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);**

**jLabel1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/xxx.png"))); // NOI18N**

**javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());**

**getContentPane().setLayout(layout);**

**layout.setHorizontalGroup(**

**layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)**

**.addGroup(layout.createSequentialGroup()**

**.addGap(107, 107, 107)**

**.addComponent(jLabel1)**

**.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))**

**);**

**layout.setVerticalGroup(**

**layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)**

**.addGroup(layout.createSequentialGroup()**

**.addGap(71, 71, 71)**

**.addComponent(jLabel1)**

**.addContainerGap(215, Short.MAX\_VALUE))**

**);**

**pack();**

**}// </editor-fold>**

**/\*\***

**\* @param args the command line arguments**

**\*/**

**public static void main(String args[]) {**

**/\* Set the Nimbus look and feel \*/**

**//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">**

**/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.**

**\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html**

**\*/**

**try {**

**for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {**

**if ("Nimbus".equals(info.getName())) {**

**javax.swing.UIManager.setLookAndFeel(info.getClassName());**

**break;**

**}**

**}**

**} catch (ClassNotFoundException ex) {**

**java.util.logging.Logger.getLogger(NewJFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);**

**} catch (InstantiationException ex) {**

**java.util.logging.Logger.getLogger(NewJFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);**

**} catch (IllegalAccessException ex) {**

**java.util.logging.Logger.getLogger(NewJFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);**

**} catch (javax.swing.UnsupportedLookAndFeelException ex) {**

**java.util.logging.Logger.getLogger(NewJFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);**

**}**

**//</editor-fold>**

**/\* Create and display the form \*/**

**java.awt.EventQueue.invokeLater(new Runnable() {**

**public void run() {**

**new NewJFrame().setVisible(true);**

**}**

**});**

**}**

**// Variables declaration - do not modify**

**private javax.swing.JLabel jLabel1;**

**// End of variables declaration**

**}**

* + 1. **Form coding of Authentication.java**

**import java.sql.Connection;**

**import java.sql.PreparedStatement;**

**import java.sql.ResultSet;**

**import javax.swing.JOptionPane;**

**import javax.swing.UIManager;**

**/\***

**\* Document: Authentication.java**

**\* Unitec Institute of Technology**

**\* @authors Natasha Bettridge, Taylor Tran and Michael Yin**

**\* Document: Contains the login from pt 1 and 2 and the actual coding watch pt 5**

**\* Go straight to the public static void section to see where i added**

**\* J tattoo in the file an add the code. Code added by Michael Yin..**

**\*/**

**/\*\***

**\***

**\* @author Natasha Bettridge, Taylor Tran, Michael Yin**

**\*/**

**public class Authentication extends javax.swing.JFrame {**

**Connection conn;**

**ResultSet rs;**

**PreparedStatement pst;**

**/\*\***

**\* Creates new form Authentication**

**\*/**

**// constructor for authentication**

**public Authentication() {**

**super("Login");**

**initComponents();**

**conn = dbconnect.connectDb();**

**}**

**/\*\***

**\* This method is called from within the constructor to initialize the form.**

**\* WARNING: Do NOT modify this code. The content of this method is always**

**\* regenerated by the Form Editor.**

**\*/**

**@SuppressWarnings("unchecked")**

**// <editor-fold defaultstate="collapsed" desc="Generated Code">**

**private void initComponents() {**

**LoginPage = new javax.swing.JPanel();**

**AccnumLP = new javax.swing.JTextField();**

**PinLP = new javax.swing.JTextField();**

**PintxtLP = new javax.swing.JLabel();**

**AccnumtxtLP = new javax.swing.JLabel();**

**Logon = new javax.swing.JButton();**

**NewAccount = new javax.swing.JButton();**

**jLabel1 = new javax.swing.JLabel();**

**setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);**

**setBackground(new java.awt.Color(255, 255, 255));**

**LoginPage.setBackground(new java.awt.Color(255, 255, 255));**

**LoginPage.setBorder(new javax.swing.border.LineBorder(new java.awt.Color(11, 56, 95), 3, true));**

**LoginPage.setAutoscrolls(true);**

**AccnumLP.addActionListener(new java.awt.event.ActionListener() {**

**public void actionPerformed(java.awt.event.ActionEvent evt) {**

**AccnumLPActionPerformed(evt);**

**}**

**});**

**PintxtLP.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N**

**PintxtLP.setText("PIN:");**

**AccnumtxtLP.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N**

**AccnumtxtLP.setText("Account Number:");**

**Logon.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N**

**Logon.setText("Log on");**

**Logon.addActionListener(new java.awt.event.ActionListener() {**

**public void actionPerformed(java.awt.event.ActionEvent evt) {**

**LogonActionPerformed(evt);**

**}**

**});**

**NewAccount.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N**

**NewAccount.setText("New Account");**

**NewAccount.addActionListener(new java.awt.event.ActionListener() {**

**public void actionPerformed(java.awt.event.ActionEvent evt) {**

**NewAccountActionPerformed(evt);**

**}**

**});**

**jLabel1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/xxx.png"))); // NOI18N**

**javax.swing.GroupLayout LoginPageLayout = new javax.swing.GroupLayout(LoginPage);**

**LoginPage.setLayout(LoginPageLayout);**

**LoginPageLayout.setHorizontalGroup(**

**LoginPageLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)**

**.addGroup(LoginPageLayout.createSequentialGroup()**

**.addGroup(LoginPageLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)**

**.addGroup(LoginPageLayout.createSequentialGroup()**

**.addGap(34, 34, 34)**

**.addGroup(LoginPageLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)**

**.addComponent(AccnumtxtLP)**

**.addComponent(PintxtLP))**

**.addGap(18, 18, 18)**

**.addGroup(LoginPageLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)**

**.addGroup(LoginPageLayout.createSequentialGroup()**

**.addComponent(Logon, javax.swing.GroupLayout.PREFERRED\_SIZE, 88, javax.swing.GroupLayout.PREFERRED\_SIZE)**

**.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)**

**.addComponent(NewAccount, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))**

**.addComponent(PinLP)**

**.addComponent(AccnumLP)))**

**.addGroup(LoginPageLayout.createSequentialGroup()**

**.addGap(76, 76, 76)**

**.addComponent(jLabel1)))**

**.addContainerGap(90, Short.MAX\_VALUE))**

**);**

**LoginPageLayout.setVerticalGroup(**

**LoginPageLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)**

**.addGroup(LoginPageLayout.createSequentialGroup()**

**.addGap(35, 35, 35)**

**.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 119, javax.swing.GroupLayout.PREFERRED\_SIZE)**

**.addGap(73, 73, 73)**

**.addGroup(LoginPageLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)**

**.addComponent(AccnumtxtLP)**

**.addGroup(LoginPageLayout.createSequentialGroup()**

**.addGap(1, 1, 1)**

**.addComponent(AccnumLP, javax.swing.GroupLayout.PREFERRED\_SIZE, 20, javax.swing.GroupLayout.PREFERRED\_SIZE)))**

**.addGap(18, 18, 18)**

**.addGroup(LoginPageLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)**

**.addComponent(PintxtLP, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)**

**.addComponent(PinLP))**

**.addGap(8, 8, 8)**

**.addGroup(LoginPageLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)**

**.addComponent(NewAccount, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)**

**.addComponent(Logon, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))**

**.addContainerGap(45, Short.MAX\_VALUE))**

**);**

**javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());**

**getContentPane().setLayout(layout);**

**layout.setHorizontalGroup(**

**layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)**

**.addComponent(LoginPage, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)**

**);**

**layout.setVerticalGroup(**

**layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)**

**.addComponent(LoginPage, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)**

**);**

**pack();**

**}// </editor-fold>**

**private void AccnumLPActionPerformed(java.awt.event.ActionEvent evt) {**

**// TODO add your handling code here:**

**}**

**// create new account**

**private void NewAccountActionPerformed(java.awt.event.ActionEvent evt) {**

**// TODO add your handling code here:**

**setVisible(false);**

**Account ob = new Account();**

**ob.setVisible(true);**

**}**

**// authenticate user details with database**

**private void LogonActionPerformed(java.awt.event.ActionEvent evt) {**

**// TODO add your handling code here:**

**String sql = "select \* from Account where Acc=? and Pin=?";**

**try{**

**pst = conn.prepareStatement(sql);**

**pst.setString(1, AccnumLP.getText());**

**pst.setString(2, PinLP.getText());**

**rs=pst.executeQuery();**

**if(rs.next()){**

**setVisible(false);**

**Loading ob = new Loading();**

**ob.setUpLoading();**

**ob.setVisible(true);**

**rs.close();**

**pst.close();**

**}else{**

**JOptionPane.showMessageDialog(null,"Incorrect login");**

**}**

**}catch(Exception e){**

**JOptionPane.showMessageDialog(null, e);**

**}finally{**

**try{**

**rs.close();**

**pst.close();**

**}catch(Exception e){**

**}**

**}**

**}**

**/\*\***

**\* @param args HEY GUYS THIS SECTION IS WHERE I ADD THE CODE INTO IT - Natasha**

**\*/**

**public static void main(String args[]) {**

**/\* Set the Nimbus look and feel \*/**

**//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">**

**/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.**

**\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html**

**\*/**

**try {**

**for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {**

**/\*if ("Nimbus".equals(info.getName())) {**

**javax.swing.UIManager.setLookAndFeel(info.getClassName());**

**break;**

**}\*/**

**UIManager.setLookAndFeel("com.jtattoo.plaf.smart.SmartLookAndFeel");**

**}**

**} catch (ClassNotFoundException ex) {**

**java.util.logging.Logger.getLogger(Authentication.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);**

**} catch (InstantiationException ex) {**

**java.util.logging.Logger.getLogger(Authentication.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);**

**} catch (IllegalAccessException ex) {**

**java.util.logging.Logger.getLogger(Authentication.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);**

**} catch (javax.swing.UnsupportedLookAndFeelException ex) {**

**java.util.logging.Logger.getLogger(Authentication.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);**

**}**

**//</editor-fold>**

**/\* Create and display the form \*/**

**java.awt.EventQueue.invokeLater(new Runnable() {**

**public void run() {**

**new Authentication().setVisible(true);**

**}**

**});**

**}**

**// Variables declaration - do not modify**

**private javax.swing.JTextField AccnumLP;**

**private javax.swing.JLabel AccnumtxtLP;**

**private javax.swing.JPanel LoginPage;**

**private javax.swing.JButton Logon;**

**private javax.swing.JButton NewAccount;**

**private javax.swing.JTextField PinLP;**

**private javax.swing.JLabel PintxtLP;**

**private javax.swing.JLabel jLabel1;**

**// End of variables declaration**

**}**

* + 1. **Form coding of DBconnect.java**

**/\***

**\* Document: MyPage.java**

**\* Unitec Institute of Technology**

**\* @authors Natasha Bettridge, Taylor Tran and Michael Yin**

**\* Document: contains database connect**

**\*/**

**/\*\***

**\***

**\* @author Natasha Bettridge, Taylor Tran, Michael Yin**

**\*/**

**import java.sql.\*;**

**import javax.swing.JOptionPane;**

**public class dbconnect {**

**Connection conn = null;**

**/\* In this section please change the path to connect to the database.**

**\*/**

**public static Connection connectDb(){**

**try{**

**Class.forName("org.sqlite.JDBC");**

**Connection conn = DriverManager.getConnection("jdbc:sqlite:bank.sqlite");**

**return conn;**

**}catch(Exception e){**

**JOptionPane.showMessageDialog(null,e);**

**}**

**return null;**

**}**

**}**

* + 1. **Form coding of UImanagertest.java**

**/\***

**\* To change this license header, choose License Headers in Project Properties.**

**\* To change this template file, choose Tools | Templates**

**\* and open the template in the editor.**

**\*/**

**import org.junit.After;**

**import org.junit.AfterClass;**

**import org.junit.Before;**

**import org.junit.BeforeClass;**

**import org.junit.Test;**

**import static org.junit.Assert.\*;**

**/\*\***

**\***

**\* @author aqua\_**

**\*/**

**public class UIManagerTest {**

**public UIManagerTest() {**

**}**

**@BeforeClass**

**public static void setUpClass() {**

**}**

**@AfterClass**

**public static void tearDownClass() {**

**}**

**@Before**

**public void setUp() {**

**}**

**@After**

**public void tearDown() {**

**}**

**/\*\***

**\* Test of setLookAndFeel method, of class UIManager.**

**\*/**

**@Test**

**public void testSetLookAndFeel() {**

**System.out.println("setLookAndFeel");**

**String comjtattooplafsmartSmartLookAndFeel = "";**

**UIManager.setLookAndFeel(comjtattooplafsmartSmartLookAndFeel);**

**// TODO review the generated test code and remove the default call to fail.**

**}**

**}**

* + 1. **Form coding of Build\_impl.XML**

**<?xml version="1.0" encoding="UTF-8"?>**

**<!--**

**\*\*\* GENERATED FROM project.xml - DO NOT EDIT \*\*\***

**\*\*\* EDIT ../build.xml INSTEAD \*\*\***

**For the purpose of easier reading the script**

**is divided into following sections:**

**- initialization**

**- compilation**

**- jar**

**- execution**

**- debugging**

**- javadoc**

**- test compilation**

**- test execution**

**- test debugging**

**- applet**

**- cleanup**

**-->**

**<project xmlns:if="ant:if" xmlns:j2seproject1="http://www.netbeans.org/ns/j2se-project/1" xmlns:j2seproject3="http://www.netbeans.org/ns/j2se-project/3" xmlns:jaxrpc="http://www.netbeans.org/ns/j2se-project/jax-rpc" xmlns:unless="ant:unless" basedir=".." default="default" name="Banking\_Management\_System-impl">**

**<fail message="Please build using Ant 1.8.0 or higher.">**

**<condition>**

**<not>**

**<antversion atleast="1.8.0"/>**

**</not>**

**</condition>**

**</fail>**

**<target depends="test,jar,javadoc" description="Build and test whole project." name="default"/>**

**<?xml version="1.0" encoding="UTF-8"?>**

**<!--**

**\*\*\* GENERATED FROM project.xml - DO NOT EDIT \*\*\***

**\*\*\* EDIT ../build.xml INSTEAD \*\*\***

**For the purpose of easier reading the script**

**is divided into following sections:**

**- initialization**

**- compilation**

**- jar**

**- execution**

**- debugging**

**- javadoc**

**- test compilation**

**- test execution**

**- test debugging**

**- applet**

**- cleanup**

**-->**

**<project xmlns:if="ant:if" xmlns:j2seproject1="http://www.netbeans.org/ns/j2se-project/1" xmlns:j2seproject3="http://www.netbeans.org/ns/j2se-project/3" xmlns:jaxrpc="http://www.netbeans.org/ns/j2se-project/jax-rpc" xmlns:unless="ant:unless" basedir=".." default="default" name="Banking\_Management\_System-impl">**

**<fail message="Please build using Ant 1.8.0 or higher.">**

**<condition>**

**<not>**

**<antversion atleast="1.8.0"/>**

**</not>**

**</condition>**

**</fail>**

**<target depends="test,jar,javadoc" description="Build and test whole project." name="default"/>**

**<?xml version="1.0" encoding="UTF-8"?>**

**<!--**

**\*\*\* GENERATED FROM project.xml - DO NOT EDIT \*\*\***

**\*\*\* EDIT ../build.xml INSTEAD \*\*\***

**For the purpose of easier reading the script**

**is divided into following sections:**

**- initialization**

**- compilation**

**- jar**

**- execution**

**- debugging**

**- javadoc**

**- test compilation**

**- test execution**

**- test debugging**

**- applet**

**- cleanup**

**-->**

**<project xmlns:if="ant:if" xmlns:j2seproject1="http://www.netbeans.org/ns/j2se-project/1" xmlns:j2seproject3="http://www.netbeans.org/ns/j2se-project/3" xmlns:jaxrpc="http://www.netbeans.org/ns/j2se-project/jax-rpc" xmlns:unless="ant:unless" basedir=".." default="default" name="Banking\_Management\_System-impl">**

**<fail message="Please build using Ant 1.8.0 or higher.">**

**<condition>**

**<not>**

**<antversion atleast="1.8.0"/>**

**</not>**

**</condition>**

**</fail>**

**<target depends="test,jar,javadoc" description="Build and test whole project." name="default"/>**

**<!--**

**======================**

**INITIALIZATION SECTION**

**======================**

**-->**

**<target name="-pre-init">**

**<!-- Empty placeholder for easier customization. -->**

**<!-- You can override this target in the ../build.xml file. -->**

**</target>**

**<target depends="-pre-init" name="-init-private">**

**<property file="nbproject/private/config.properties"/>**

**<property file="nbproject/private/configs/${config}.properties"/>**

**<property file="nbproject/private/private.properties"/>**

**</target>**

**<target depends="-pre-init,-init-private" name="-init-user">**

**<property file="${user.properties.file}"/>**

**<!-- The two properties below are usually overridden -->**

**<!-- by the active platform. Just a fallback. -->**

**<property name="default.javac.source" value="1.6"/>**

**<property name="default.javac.target" value="1.6"/>**

**</target>**

**<target depends="-pre-init,-init-private,-init-user" name="-init-project">**

**<property file="nbproject/configs/${config}.properties"/>**

**<property file="nbproject/project.properties"/>**

**</target>**

**<target name="-init-modules-supported">**

**<condition property="modules.supported.internal" value="true">**

**<not>**

**<matches pattern="1\.[0-8](\..\*)?" string="${javac.source}"/>**

**</not>**

**</condition>**

**</target>**

**<target depends="-init-modules-supported" if="modules.supported.internal" name="-init-macrodef-modulename">**

**<macrodef name="modulename" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute name="property"/>**

**<attribute name="sourcepath"/>**

**<sequential>**

**<loadresource property="@{property}" quiet="true">**

**<javaresource classpath="@{sourcepath}" name="module-info.java" parentFirst="false"/>**

**<filterchain>**

**<stripjavacomments/>**

**<linecontainsregexp>**

**<regexp pattern="module .\* \{"/>**

**</linecontainsregexp>**

**<tokenfilter>**

**<linetokenizer/>**

**<replaceregex flags="s" pattern="(\s\*module\s+)(\S\*)(\s\*\{.\*)" replace="\2"/>**

**</tokenfilter>**

**<striplinebreaks/>**

**</filterchain>**

**</loadresource>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-modules-supported,-init-macrodef-modulename" if="modules.supported.internal" name="-init-source-module-properties">**

**<fail message="Java 9 support requires Ant 1.10.0 or higher.">**

**<condition>**

**<not>**

**<antversion atleast="1.10.0"/>**

**</not>**

**</condition>**

**</fail>**

**<j2seproject3:modulename property="module.name" sourcepath="${src.dir}"/>**

**<condition property="named.module.internal">**

**<and>**

**<isset property="module.name"/>**

**<length length="0" string="${module.name}" when="greater"/>**

**</and>**

**</condition>**

**<condition property="unnamed.module.internal">**

**<not>**

**<isset property="named.module.internal"/>**

**</not>**

**</condition>**

**<property name="javac.modulepath" value=""/>**

**<property name="run.modulepath" value="${javac.modulepath}"/>**

**<property name="module.build.classes.dir" value="${build.classes.dir}"/>**

**<property name="debug.modulepath" value="${run.modulepath}"/>**

**<property name="javac.upgrademodulepath" value=""/>**

**<property name="run.upgrademodulepath" value="${javac.upgrademodulepath}"/>**

**<condition else="" property="javac.systemmodulepath.cmd.line.arg" value="--system '${javac.systemmodulepath}'">**

**<and>**

**<isset property="javac.systemmodulepath"/>**

**<length length="0" string="${javac.systemmodulepath}" when="greater"/>**

**</and>**

**</condition>**

**<property name="dist.jlink.dir" value="${dist.dir}/jlink"/>**

**<property name="dist.jlink.output" value="${dist.jlink.dir}/${application.title}"/>**

**<property name="module.name" value=""/>**

**</target>**

**<target depends="-pre-init,-init-private,-init-user,-init-project,-init-macrodef-property,-init-modules-supported" name="-do-init">**

**<property name="platform.java" value="${java.home}/bin/java"/>**

**<available file="${manifest.file}" property="manifest.available"/>**

**<condition property="splashscreen.available">**

**<and>**

**<not>**

**<equals arg1="${application.splash}" arg2="" trim="true"/>**

**</not>**

**<available file="${application.splash}"/>**

**</and>**

**</condition>**

**<condition property="main.class.available">**

**<and>**

**<isset property="main.class"/>**

**<not>**

**<equals arg1="${main.class}" arg2="" trim="true"/>**

**</not>**

**</and>**

**</condition>**

**<condition property="profile.available">**

**<and>**

**<isset property="javac.profile"/>**

**<length length="0" string="${javac.profile}" when="greater"/>**

**<not>**

**<matches pattern="1\.[0-7](\..\*)?" string="${javac.source}"/>**

**</not>**

**</and>**

**</condition>**

**<condition property="do.archive">**

**<or>**

**<not>**

**<istrue value="${jar.archive.disabled}"/>**

**</not>**

**<istrue value="${not.archive.disabled}"/>**

**</or>**

**</condition>**

**<condition property="do.archive+manifest.available">**

**<and>**

**<isset property="manifest.available"/>**

**<istrue value="${do.archive}"/>**

**</and>**

**</condition>**

**<condition property="do.archive+main.class.available">**

**<and>**

**<isset property="main.class.available"/>**

**<istrue value="${do.archive}"/>**

**</and>**

**</condition>**

**<condition property="do.archive+splashscreen.available">**

**<and>**

**<isset property="splashscreen.available"/>**

**<istrue value="${do.archive}"/>**

**</and>**

**</condition>**

**<condition property="do.archive+profile.available">**

**<and>**

**<isset property="profile.available"/>**

**<istrue value="${do.archive}"/>**

**</and>**

**</condition>**

**<condition property="have.tests">**

**<or>**

**<available file="${test.src.dir}"/>**

**</or>**

**</condition>**

**<condition property="have.sources">**

**<or>**

**<available file="${src.dir}"/>**

**</or>**

**</condition>**

**<condition property="netbeans.home+have.tests">**

**<and>**

**<isset property="netbeans.home"/>**

**<isset property="have.tests"/>**

**</and>**

**</condition>**

**<condition property="no.javadoc.preview">**

**<and>**

**<isset property="javadoc.preview"/>**

**<isfalse value="${javadoc.preview}"/>**

**</and>**

**</condition>**

**<property name="run.jvmargs" value=""/>**

**<property name="run.jvmargs.ide" value=""/>**

**<property name="javac.compilerargs" value=""/>**

**<property name="work.dir" value="${basedir}"/>**

**<condition property="no.deps">**

**<and>**

**<istrue value="${no.dependencies}"/>**

**</and>**

**</condition>**

**<property name="javac.debug" value="true"/>**

**<property name="javadoc.preview" value="true"/>**

**<property name="application.args" value=""/>**

**<property name="source.encoding" value="${file.encoding}"/>**

**<property name="runtime.encoding" value="${source.encoding}"/>**

**<property name="manifest.encoding" value="${source.encoding}"/>**

**<condition property="javadoc.encoding.used" value="${javadoc.encoding}">**

**<and>**

**<isset property="javadoc.encoding"/>**

**<not>**

**<equals arg1="${javadoc.encoding}" arg2=""/>**

**</not>**

**</and>**

**</condition>**

**<property name="javadoc.encoding.used" value="${source.encoding}"/>**

**<property name="includes" value="\*\*"/>**

**<property name="excludes" value=""/>**

**<property name="do.depend" value="false"/>**

**<condition property="do.depend.true">**

**<istrue value="${do.depend}"/>**

**</condition>**

**<path id="endorsed.classpath.path" path="${endorsed.classpath}"/>**

**<condition else="" property="endorsed.classpath.cmd.line.arg" value="-Xbootclasspath/p:'${toString:endorsed.classpath.path}'">**

**<and>**

**<isset property="endorsed.classpath"/>**

**<not>**

**<equals arg1="${endorsed.classpath}" arg2="" trim="true"/>**

**</not>**

**</and>**

**</condition>**

**<condition else="" property="javac.profile.cmd.line.arg" value="-profile ${javac.profile}">**

**<isset property="profile.available"/>**

**</condition>**

**<condition else="false" property="jdkBug6558476">**

**<and>**

**<matches pattern="1\.[56]" string="${java.specification.version}"/>**

**<not>**

**<os family="unix"/>**

**</not>**

**</and>**

**</condition>**

**<condition else="false" property="javac.fork">**

**<or>**

**<istrue value="${jdkBug6558476}"/>**

**<istrue value="${javac.external.vm}"/>**

**</or>**

**</condition>**

**<property name="jar.index" value="false"/>**

**<property name="jar.index.metainf" value="${jar.index}"/>**

**<property name="copylibs.rebase" value="true"/>**

**<available file="${meta.inf.dir}/persistence.xml" property="has.persistence.xml"/>**

**<condition property="junit.available">**

**<or>**

**<available classname="org.junit.Test" classpath="${run.test.classpath}"/>**

**<available classname="junit.framework.Test" classpath="${run.test.classpath}"/>**

**</or>**

**</condition>**

**<condition property="testng.available">**

**<available classname="org.testng.annotations.Test" classpath="${run.test.classpath}"/>**

**</condition>**

**<condition property="junit+testng.available">**

**<and>**

**<istrue value="${junit.available}"/>**

**<istrue value="${testng.available}"/>**

**</and>**

**</condition>**

**<condition else="testng" property="testng.mode" value="mixed">**

**<istrue value="${junit+testng.available}"/>**

**</condition>**

**<condition else="" property="testng.debug.mode" value="-mixed">**

**<istrue value="${junit+testng.available}"/>**

**</condition>**

**<property name="java.failonerror" value="true"/>**

**</target>**

**<target name="-post-init">**

**<!-- Empty placeholder for easier customization. -->**

**<!-- You can override this target in the ../build.xml file. -->**

**</target>**

**<target depends="-pre-init,-init-private,-init-user,-init-project,-do-init" name="-init-check">**

**<fail unless="src.dir">Must set src.dir</fail>**

**<fail unless="test.src.dir">Must set test.src.dir</fail>**

**<fail unless="build.dir">Must set build.dir</fail>**

**<fail unless="dist.dir">Must set dist.dir</fail>**

**<fail unless="build.classes.dir">Must set build.classes.dir</fail>**

**<fail unless="dist.javadoc.dir">Must set dist.javadoc.dir</fail>**

**<fail unless="build.test.classes.dir">Must set build.test.classes.dir</fail>**

**<fail unless="build.test.results.dir">Must set build.test.results.dir</fail>**

**<fail unless="build.classes.excludes">Must set build.classes.excludes</fail>**

**<fail unless="dist.jar">Must set dist.jar</fail>**

**</target>**

**<target name="-init-macrodef-property">**

**<macrodef name="property" uri="http://www.netbeans.org/ns/j2se-project/1">**

**<attribute name="name"/>**

**<attribute name="value"/>**

**<sequential>**

**<property name="@{name}" value="${@{value}}"/>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-ap-cmdline-properties,-init-source-module-properties" if="modules.supported.internal" name="-init-macrodef-javac-with-module">**

**<macrodef name="javac" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${src.dir}" name="srcdir"/>**

**<attribute default="${build.classes.dir}" name="destdir"/>**

**<attribute default="${javac.classpath}" name="classpath"/>**

**<attribute default="${javac.modulepath}" name="modulepath"/>**

**<attribute default="${javac.upgrademodulepath}" name="upgrademodulepath"/>**

**<attribute default="${javac.processorpath}" name="processorpath"/>**

**<attribute default="${javac.processormodulepath}" name="processormodulepath"/>**

**<attribute default="${build.generated.sources.dir}/ap-source-output" name="apgeneratedsrcdir"/>**

**<attribute default="${includes}" name="includes"/>**

**<attribute default="${excludes}" name="excludes"/>**

**<attribute default="${javac.debug}" name="debug"/>**

**<attribute default="${empty.dir}" name="sourcepath" unless:set="named.module.internal"/>**

**<attribute default="${src.dir}" if:set="named.module.internal" name="sourcepath"/>**

**<attribute default="${empty.dir}" name="gensrcdir"/>**

**<element name="customize" optional="true"/>**

**<sequential>**

**<condition property="warn.excludes.internal">**

**<and>**

**<isset property="named.module.internal"/>**

**<length length="0" string="@{excludes}" trim="true" when="greater"/>**

**</and>**

**</condition>**

**<echo if:set="warn.excludes.internal" level="warning" message="The javac excludes are not supported in the JDK 9 Named Module."/>**

**<property location="${build.dir}/empty" name="empty.dir"/>**

**<mkdir dir="${empty.dir}"/>**

**<mkdir dir="@{apgeneratedsrcdir}"/>**

**<condition property="processormodulepath.set">**

**<resourcecount count="0" when="greater">**

**<path>**

**<pathelement path="@{processormodulepath}"/>**

**</path>**

**</resourcecount>**

**</condition>**

**<javac debug="@{debug}" deprecation="${javac.deprecation}" destdir="@{destdir}" encoding="${source.encoding}" excludes="@{excludes}" fork="${javac.fork}" includeantruntime="false" includes="@{includes}" source="${javac.source}" sourcepath="@{sourcepath}" srcdir="@{srcdir}" target="${javac.target}" tempdir="${java.io.tmpdir}">**

**<src>**

**<dirset dir="@{gensrcdir}" erroronmissingdir="false">**

**<include name="\*"/>**

**</dirset>**

**</src>**

**<classpath>**

**<path path="@{classpath}"/>**

**</classpath>**

**<modulepath>**

**<path path="@{modulepath}"/>**

**</modulepath>**

**<upgrademodulepath>**

**<path path="@{upgrademodulepath}"/>**

**</upgrademodulepath>**

**<compilerarg line="${javac.systemmodulepath.cmd.line.arg}"/>**

**<compilerarg line="${javac.profile.cmd.line.arg}"/>**

**<compilerarg line="${javac.compilerargs}"/>**

**<compilerarg if:set="processormodulepath.set" value="--processor-module-path"/>**

**<compilerarg if:set="processormodulepath.set" path="@{processormodulepath}"/>**

**<compilerarg unless:set="processormodulepath.set" value="-processorpath"/>**

**<compilerarg path="@{processorpath}:${empty.dir}" unless:set="processormodulepath.set"/>**

**<compilerarg line="${ap.processors.internal}"/>**

**<compilerarg line="${annotation.processing.processor.options}"/>**

**<compilerarg value="-s"/>**

**<compilerarg path="@{apgeneratedsrcdir}"/>**

**<compilerarg line="${ap.proc.none.internal}"/>**

**<customize/>**

**</javac>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-ap-cmdline-properties,-init-source-module-properties" if="ap.supported.internal" name="-init-macrodef-javac-with-processors" unless="modules.supported.internal">**

**<macrodef name="javac" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${src.dir}" name="srcdir"/>**

**<attribute default="${build.classes.dir}" name="destdir"/>**

**<attribute default="${javac.classpath}" name="classpath"/>**

**<attribute default="${javac.modulepath}" name="modulepath"/>**

**<attribute default="${javac.upgrademodulepath}" name="upgrademodulepath"/>**

**<attribute default="${javac.processorpath}" name="processorpath"/>**

**<attribute default="${build.generated.sources.dir}/ap-source-output" name="apgeneratedsrcdir"/>**

**<attribute default="${includes}" name="includes"/>**

**<attribute default="${excludes}" name="excludes"/>**

**<attribute default="${javac.debug}" name="debug"/>**

**<attribute default="${empty.dir}" name="sourcepath"/>**

**<attribute default="${empty.dir}" name="gensrcdir"/>**

**<element name="customize" optional="true"/>**

**<sequential>**

**<property location="${build.dir}/empty" name="empty.dir"/>**

**<mkdir dir="${empty.dir}"/>**

**<mkdir dir="@{apgeneratedsrcdir}"/>**

**<javac debug="@{debug}" deprecation="${javac.deprecation}" destdir="@{destdir}" encoding="${source.encoding}" excludes="@{excludes}" fork="${javac.fork}" includeantruntime="false" includes="@{includes}" source="${javac.source}" sourcepath="@{sourcepath}" srcdir="@{srcdir}" target="${javac.target}" tempdir="${java.io.tmpdir}">**

**<src>**

**<dirset dir="@{gensrcdir}" erroronmissingdir="false">**

**<include name="\*"/>**

**</dirset>**

**</src>**

**<classpath>**

**<path path="@{classpath}"/>**

**</classpath>**

**<compilerarg line="${endorsed.classpath.cmd.line.arg}"/>**

**<compilerarg line="${javac.profile.cmd.line.arg}"/>**

**<compilerarg line="${javac.compilerargs}"/>**

**<compilerarg value="-processorpath"/>**

**<compilerarg path="@{processorpath}:${empty.dir}"/>**

**<compilerarg line="${ap.processors.internal}"/>**

**<compilerarg line="${annotation.processing.processor.options}"/>**

**<compilerarg value="-s"/>**

**<compilerarg path="@{apgeneratedsrcdir}"/>**

**<compilerarg line="${ap.proc.none.internal}"/>**

**<customize/>**

**</javac>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-ap-cmdline-properties,-init-source-module-properties" name="-init-macrodef-javac-without-processors" unless="ap.supported.internal">**

**<macrodef name="javac" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${src.dir}" name="srcdir"/>**

**<attribute default="${build.classes.dir}" name="destdir"/>**

**<attribute default="${javac.classpath}" name="classpath"/>**

**<attribute default="${javac.modulepath}" name="modulepath"/>**

**<attribute default="${javac.upgrademodulepath}" name="upgrademodulepath"/>**

**<attribute default="${javac.processorpath}" name="processorpath"/>**

**<attribute default="${build.generated.sources.dir}/ap-source-output" name="apgeneratedsrcdir"/>**

**<attribute default="${includes}" name="includes"/>**

**<attribute default="${excludes}" name="excludes"/>**

**<attribute default="${javac.debug}" name="debug"/>**

**<attribute default="${empty.dir}" name="sourcepath"/>**

**<attribute default="${empty.dir}" name="gensrcdir"/>**

**<element name="customize" optional="true"/>**

**<sequential>**

**<property location="${build.dir}/empty" name="empty.dir"/>**

**<mkdir dir="${empty.dir}"/>**

**<javac debug="@{debug}" deprecation="${javac.deprecation}" destdir="@{destdir}" encoding="${source.encoding}" excludes="@{excludes}" fork="${javac.fork}" includeantruntime="false" includes="@{includes}" source="${javac.source}" sourcepath="@{sourcepath}" srcdir="@{srcdir}" target="${javac.target}" tempdir="${java.io.tmpdir}">**

**<src>**

**<dirset dir="@{gensrcdir}" erroronmissingdir="false">**

**<include name="\*"/>**

**</dirset>**

**</src>**

**<classpath>**

**<path path="@{classpath}"/>**

**</classpath>**

**<compilerarg line="${endorsed.classpath.cmd.line.arg}"/>**

**<compilerarg line="${javac.profile.cmd.line.arg}"/>**

**<compilerarg line="${javac.compilerargs}"/>**

**<customize/>**

**</javac>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-macrodef-javac-with-module,-init-macrodef-javac-with-processors,-init-macrodef-javac-without-processors" name="-init-macrodef-javac">**

**<macrodef name="depend" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${src.dir}" name="srcdir"/>**

**<attribute default="${build.classes.dir}" name="destdir"/>**

**<attribute default="${javac.classpath}" name="classpath"/>**

**<sequential>**

**<depend cache="${build.dir}/depcache" destdir="@{destdir}" excludes="${excludes}" includes="${includes}" srcdir="@{srcdir}">**

**<classpath>**

**<path path="@{classpath}"/>**

**</classpath>**

**</depend>**

**</sequential>**

**</macrodef>**

**<macrodef name="force-recompile" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${build.classes.dir}" name="destdir"/>**

**<sequential>**

**<fail unless="javac.includes">Must set javac.includes</fail>**

**<pathconvert pathsep="${line.separator}" property="javac.includes.binary">**

**<path>**

**<filelist dir="@{destdir}" files="${javac.includes}"/>**

**</path>**

**<globmapper from="\*.java" to="\*.class"/>**

**</pathconvert>**

**<tempfile deleteonexit="true" property="javac.includesfile.binary"/>**

**<echo file="${javac.includesfile.binary}" message="${javac.includes.binary}"/>**

**<delete>**

**<files includesfile="${javac.includesfile.binary}"/>**

**</delete>**

**<delete>**

**<fileset file="${javac.includesfile.binary}"/>**

**</delete>**

**</sequential>**

**</macrodef>**

**</target>**

**<target if="${junit.available}" name="-init-macrodef-junit-init">**

**<condition else="false" property="nb.junit.batch" value="true">**

**<and>**

**<istrue value="${junit.available}"/>**

**<not>**

**<isset property="test.method"/>**

**</not>**

**</and>**

**</condition>**

**<condition else="false" property="nb.junit.single" value="true">**

**<and>**

**<istrue value="${junit.available}"/>**

**<isset property="test.method"/>**

**</and>**

**</condition>**

**</target>**

**<target name="-init-test-properties">**

**<property name="test.binaryincludes" value="&lt;nothing&gt;"/>**

**<property name="test.binarytestincludes" value=""/>**

**<property name="test.binaryexcludes" value=""/>**

**</target>**

**<target depends="-init-modules-supported" if="modules.supported.internal" name="-init-macrodef-junit-prototype-with-module">**

**<macrodef name="junit-prototype" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${includes}" name="includes"/>**

**<attribute default="${excludes}" name="excludes"/>**

**<element name="customizePrototype" optional="true"/>**

**<sequential>**

**<property name="junit.forkmode" value="perTest"/>**

**<junit dir="${work.dir}" errorproperty="tests.failed" failureproperty="tests.failed" fork="true" forkmode="${junit.forkmode}" showoutput="true" tempdir="${build.dir}">**

**<syspropertyset>**

**<propertyref prefix="test-sys-prop."/>**

**<mapper from="test-sys-prop.\*" to="\*" type="glob"/>**

**</syspropertyset>**

**<classpath>**

**<path path="${run.test.classpath}"/>**

**</classpath>**

**<modulepath>**

**<path path="${run.test.modulepath}"/>**

**</modulepath>**

**<formatter type="brief" usefile="false"/>**

**<formatter type="xml"/>**

**<jvmarg line="${endorsed.classpath.cmd.line.arg}"/>**

**<jvmarg value="-ea"/>**

**<jvmarg line="${run.test.jvmargs}"/>**

**<customizePrototype/>**

**</junit>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-modules-supported" name="-init-macrodef-junit-prototype-without-module" unless="modules.supported.internal">**

**<macrodef name="junit-prototype" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${includes}" name="includes"/>**

**<attribute default="${excludes}" name="excludes"/>**

**<element name="customizePrototype" optional="true"/>**

**<sequential>**

**<property name="junit.forkmode" value="perTest"/>**

**<junit dir="${work.dir}" errorproperty="tests.failed" failureproperty="tests.failed" fork="true" forkmode="${junit.forkmode}" showoutput="true" tempdir="${build.dir}">**

**<syspropertyset>**

**<propertyref prefix="test-sys-prop."/>**

**<mapper from="test-sys-prop.\*" to="\*" type="glob"/>**

**</syspropertyset>**

**<classpath>**

**<path path="${run.test.classpath}"/>**

**</classpath>**

**<formatter type="brief" usefile="false"/>**

**<formatter type="xml"/>**

**<jvmarg line="${endorsed.classpath.cmd.line.arg}"/>**

**<jvmarg value="-ea"/>**

**<customizePrototype/>**

**</junit>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-test-properties,-init-macrodef-junit-prototype-with-module,-init-macrodef-junit-prototype-without-module" if="${nb.junit.single}" name="-init-macrodef-junit-single" unless="${nb.junit.batch}">**

**<macrodef name="junit" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${includes}" name="includes"/>**

**<attribute default="${excludes}" name="excludes"/>**

**<attribute default="\*\*" name="testincludes"/>**

**<attribute default="" name="testmethods"/>**

**<element name="customize" optional="true"/>**

**<sequential>**

**<j2seproject3:junit-prototype>**

**<customizePrototype>**

**<test methods="@{testmethods}" name="@{testincludes}" todir="${build.test.results.dir}"/>**

**<customize/>**

**</customizePrototype>**

**</j2seproject3:junit-prototype>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-test-properties,-init-macrodef-junit-prototype-with-module,-init-macrodef-junit-prototype-without-module" if="${nb.junit.batch}" name="-init-macrodef-junit-batch" unless="${nb.junit.single}">**

**<macrodef name="junit" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${includes}" name="includes"/>**

**<attribute default="${excludes}" name="excludes"/>**

**<attribute default="\*\*" name="testincludes"/>**

**<attribute default="" name="testmethods"/>**

**<element name="customize" optional="true"/>**

**<sequential>**

**<j2seproject3:junit-prototype>**

**<customizePrototype>**

**<batchtest todir="${build.test.results.dir}">**

**<fileset dir="${test.src.dir}" excludes="@{excludes},${excludes}" includes="@{includes}">**

**<filename name="@{testincludes}"/>**

**</fileset>**

**<fileset dir="${build.test.classes.dir}" excludes="@{excludes},${excludes},${test.binaryexcludes}" includes="${test.binaryincludes}">**

**<filename name="${test.binarytestincludes}"/>**

**</fileset>**

**</batchtest>**

**<customize/>**

**</customizePrototype>**

**</j2seproject3:junit-prototype>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-macrodef-junit-init,-init-macrodef-junit-single, -init-macrodef-junit-batch" if="${junit.available}" name="-init-macrodef-junit"/>**

**<target if="${testng.available}" name="-init-macrodef-testng">**

**<macrodef name="testng" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${includes}" name="includes"/>**

**<attribute default="${excludes}" name="excludes"/>**

**<attribute default="\*\*" name="testincludes"/>**

**<attribute default="" name="testmethods"/>**

**<element name="customize" optional="true"/>**

**<sequential>**

**<condition else="" property="testng.methods.arg" value="@{testincludes}.@{testmethods}">**

**<isset property="test.method"/>**

**</condition>**

**<union id="test.set">**

**<fileset dir="${test.src.dir}" excludes="@{excludes},\*\*/\*.xml,${excludes}" includes="@{includes}">**

**<filename name="@{testincludes}"/>**

**</fileset>**

**</union>**

**<taskdef classname="org.testng.TestNGAntTask" classpath="${run.test.classpath}" name="testng"/>**

**<testng classfilesetref="test.set" failureProperty="tests.failed" listeners="org.testng.reporters.VerboseReporter" methods="${testng.methods.arg}" mode="${testng.mode}" outputdir="${build.test.results.dir}" suitename="Banking\_Management\_System" testname="TestNG tests" workingDir="${work.dir}">**

**<xmlfileset dir="${build.test.classes.dir}" includes="@{testincludes}"/>**

**<propertyset>**

**<propertyref prefix="test-sys-prop."/>**

**<mapper from="test-sys-prop.\*" to="\*" type="glob"/>**

**</propertyset>**

**<classpath>**

**<path path="${run.test.classpath}"/>**

**</classpath>**

**<jvmarg line="${endorsed.classpath.cmd.line.arg}"/>**

**<customize/>**

**</testng>**

**</sequential>**

**</macrodef>**

**</target>**

**<target name="-init-macrodef-test-impl">**

**<macrodef name="test-impl" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${includes}" name="includes"/>**

**<attribute default="${excludes}" name="excludes"/>**

**<attribute default="\*\*" name="testincludes"/>**

**<attribute default="" name="testmethods"/>**

**<element implicit="true" name="customize" optional="true"/>**

**<sequential>**

**<echo>No tests executed.</echo>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-macrodef-junit" if="${junit.available}" name="-init-macrodef-junit-impl">**

**<macrodef name="test-impl" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${includes}" name="includes"/>**

**<attribute default="${excludes}" name="excludes"/>**

**<attribute default="\*\*" name="testincludes"/>**

**<attribute default="" name="testmethods"/>**

**<element implicit="true" name="customize" optional="true"/>**

**<sequential>**

**<j2seproject3:junit excludes="@{excludes}" includes="@{includes}" testincludes="@{testincludes}" testmethods="@{testmethods}">**

**<customize/>**

**</j2seproject3:junit>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-macrodef-testng" if="${testng.available}" name="-init-macrodef-testng-impl">**

**<macrodef name="test-impl" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${includes}" name="includes"/>**

**<attribute default="${excludes}" name="excludes"/>**

**<attribute default="\*\*" name="testincludes"/>**

**<attribute default="" name="testmethods"/>**

**<element implicit="true" name="customize" optional="true"/>**

**<sequential>**

**<j2seproject3:testng excludes="@{excludes}" includes="@{includes}" testincludes="@{testincludes}" testmethods="@{testmethods}">**

**<customize/>**

**</j2seproject3:testng>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-macrodef-test-impl,-init-macrodef-junit-impl,-init-macrodef-testng-impl" name="-init-macrodef-test">**

**<macrodef name="test" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${includes}" name="includes"/>**

**<attribute default="${excludes}" name="excludes"/>**

**<attribute default="\*\*" name="testincludes"/>**

**<attribute default="" name="testmethods"/>**

**<sequential>**

**<j2seproject3:test-impl excludes="@{excludes}" includes="@{includes}" testincludes="@{testincludes}" testmethods="@{testmethods}">**

**<customize>**

**<jvmarg line="${run.jvmargs}"/>**

**<jvmarg line="${run.jvmargs.ide}"/>**

**</customize>**

**</j2seproject3:test-impl>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-macrodef-junit" if="${junit.available}" name="-init-macrodef-junit-debug-impl">**

**<macrodef name="test-debug-impl" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${includes}" name="includes"/>**

**<attribute default="${excludes}" name="excludes"/>**

**<attribute default="\*\*" name="testincludes"/>**

**<attribute default="" name="testmethods"/>**

**<element name="customizeDebuggee" optional="true"/>**

**<sequential>**

**<j2seproject3:junit excludes="@{excludes}" includes="@{includes}" testincludes="@{testincludes}" testmethods="@{testmethods}">**

**<customize>**

**<jvmarg value="-agentlib:jdwp=transport=${debug-transport},address=${jpda.address}"/>**

**<customizeDebuggee/>**

**</customize>**

**</j2seproject3:junit>**

**</sequential>**

**</macrodef>**

**</target>**

**<target if="${testng.available}" name="-init-macrodef-testng-debug">**

**<macrodef name="testng-debug" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${main.class}" name="testClass"/>**

**<attribute default="" name="testMethod"/>**

**<element name="customize2" optional="true"/>**

**<sequential>**

**<condition else="-testclass @{testClass}" property="test.class.or.method" value="-methods @{testClass}.@{testMethod}">**

**<isset property="test.method"/>**

**</condition>**

**<condition else="-suitename Banking\_Management\_System -testname @{testClass} ${test.class.or.method}" property="testng.cmd.args" value="@{testClass}">**

**<matches pattern=".\*\.xml" string="@{testClass}"/>**

**</condition>**

**<delete dir="${build.test.results.dir}" quiet="true"/>**

**<mkdir dir="${build.test.results.dir}"/>**

**<j2seproject3:debug classname="org.testng.TestNG" classpath="${debug.test.classpath}">**

**<customizeDebuggee>**

**<customize2/>**

**<jvmarg value="-ea"/>**

**<arg line="${testng.debug.mode}"/>**

**<arg line="-d ${build.test.results.dir}"/>**

**<arg line="-listener org.testng.reporters.VerboseReporter"/>**

**<arg line="${testng.cmd.args}"/>**

**</customizeDebuggee>**

**</j2seproject3:debug>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-macrodef-testng-debug" if="${testng.available}" name="-init-macrodef-testng-debug-impl">**

**<macrodef name="testng-debug-impl" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${main.class}" name="testClass"/>**

**<attribute default="" name="testMethod"/>**

**<element implicit="true" name="customize2" optional="true"/>**

**<sequential>**

**<j2seproject3:testng-debug testClass="@{testClass}" testMethod="@{testMethod}">**

**<customize2/>**

**</j2seproject3:testng-debug>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-macrodef-junit-debug-impl" if="${junit.available}" name="-init-macrodef-test-debug-junit">**

**<macrodef name="test-debug" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${includes}" name="includes"/>**

**<attribute default="${excludes}" name="excludes"/>**

**<attribute default="\*\*" name="testincludes"/>**

**<attribute default="" name="testmethods"/>**

**<attribute default="${main.class}" name="testClass"/>**

**<attribute default="" name="testMethod"/>**

**<sequential>**

**<j2seproject3:test-debug-impl excludes="@{excludes}" includes="@{includes}" testincludes="@{testincludes}" testmethods="@{testmethods}">**

**<customizeDebuggee>**

**<jvmarg line="${run.jvmargs}"/>**

**<jvmarg line="${run.jvmargs.ide}"/>**

**</customizeDebuggee>**

**</j2seproject3:test-debug-impl>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-macrodef-testng-debug-impl" if="${testng.available}" name="-init-macrodef-test-debug-testng">**

**<macrodef name="test-debug" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${includes}" name="includes"/>**

**<attribute default="${excludes}" name="excludes"/>**

**<attribute default="\*\*" name="testincludes"/>**

**<attribute default="" name="testmethods"/>**

**<attribute default="${main.class}" name="testClass"/>**

**<attribute default="" name="testMethod"/>**

**<sequential>**

**<j2seproject3:testng-debug-impl testClass="@{testClass}" testMethod="@{testMethod}">**

**<customize2>**

**<syspropertyset>**

**<propertyref prefix="test-sys-prop."/>**

**<mapper from="test-sys-prop.\*" to="\*" type="glob"/>**

**</syspropertyset>**

**</customize2>**

**</j2seproject3:testng-debug-impl>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-macrodef-test-debug-junit,-init-macrodef-test-debug-testng" name="-init-macrodef-test-debug"/>**

**<!--**

**pre NB7.2 profiling section; consider it deprecated**

**-->**

**<target depends="-profile-pre-init, init, -profile-post-init, -profile-init-macrodef-profile, -profile-init-check" if="profiler.info.jvmargs.agent" name="profile-init"/>**

**<target if="profiler.info.jvmargs.agent" name="-profile-pre-init">**

**<!-- Empty placeholder for easier customization. -->**

**<!-- You can override this target in the ../build.xml file. -->**

**</target>**

**<target if="profiler.info.jvmargs.agent" name="-profile-post-init">**

**<!-- Empty placeholder for easier customization. -->**

**<!-- You can override this target in the ../build.xml file. -->**

**</target>**

**<target if="profiler.info.jvmargs.agent" name="-profile-init-macrodef-profile">**

**<macrodef name="resolve">**

**<attribute name="name"/>**

**<attribute name="value"/>**

**<sequential>**

**<property name="@{name}" value="${env.@{value}}"/>**

**</sequential>**

**</macrodef>**

**<macrodef name="profile">**

**<attribute default="${main.class}" name="classname"/>**

**<element name="customize" optional="true"/>**

**<sequential>**

**<property environment="env"/>**

**<resolve name="profiler.current.path" value="${profiler.info.pathvar}"/>**

**<java classname="@{classname}" dir="${profiler.info.dir}" failonerror="${java.failonerror}" fork="true" jvm="${profiler.info.jvm}">**

**<jvmarg line="${endorsed.classpath.cmd.line.arg}"/>**

**<jvmarg value="${profiler.info.jvmargs.agent}"/>**

**<jvmarg line="${profiler.info.jvmargs}"/>**

**<env key="${profiler.info.pathvar}" path="${profiler.info.agentpath}:${profiler.current.path}"/>**

**<arg line="${application.args}"/>**

**<classpath>**

**<path path="${run.classpath}"/>**

**</classpath>**

**<syspropertyset>**

**<propertyref prefix="run-sys-prop."/>**

**<mapper from="run-sys-prop.\*" to="\*" type="glob"/>**

**</syspropertyset>**

**<customize/>**

**</java>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-profile-pre-init, init, -profile-post-init, -profile-init-macrodef-profile" if="profiler.info.jvmargs.agent" name="-profile-init-check">**

**<fail unless="profiler.info.jvm">Must set JVM to use for profiling in profiler.info.jvm</fail>**

**<fail unless="profiler.info.jvmargs.agent">Must set profiler agent JVM arguments in profiler.info.jvmargs.agent</fail>**

**</target>**

**<!--**

**end of pre NB7.2 profiling section**

**-->**

**<target depends="-init-debug-args" name="-init-macrodef-nbjpda">**

**<macrodef name="nbjpdastart" uri="http://www.netbeans.org/ns/j2se-project/1">**

**<attribute default="${main.class}" name="name"/>**

**<attribute default="${debug.modulepath}" name="modulepath"/>**

**<attribute default="${debug.classpath}" name="classpath"/>**

**<attribute default="" name="stopclassname"/>**

**<sequential>**

**<nbjpdastart addressproperty="jpda.address" name="@{name}" stopclassname="@{stopclassname}" transport="${debug-transport}">**

**<modulepath>**

**<path path="@{modulepath}"/>**

**</modulepath>**

**<classpath>**

**<path path="@{classpath}"/>**

**</classpath>**

**</nbjpdastart>**

**</sequential>**

**</macrodef>**

**<macrodef name="nbjpdareload" uri="http://www.netbeans.org/ns/j2se-project/1">**

**<attribute default="${build.classes.dir}" name="dir"/>**

**<sequential>**

**<nbjpdareload>**

**<fileset dir="@{dir}" includes="${fix.classes}">**

**<include name="${fix.includes}\*.class"/>**

**</fileset>**

**</nbjpdareload>**

**</sequential>**

**</macrodef>**

**</target>**

**<target name="-init-debug-args">**

**<condition else="dt\_socket" property="debug-transport-by-os" value="dt\_shmem">**

**<os family="windows"/>**

**</condition>**

**<condition else="${debug-transport-by-os}" property="debug-transport" value="${debug.transport}">**

**<isset property="debug.transport"/>**

**</condition>**

**</target>**

**<target depends="-init-debug-args" name="-init-macrodef-debug">**

**<macrodef name="debug" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${module.name}" name="modulename"/>**

**<attribute default="${main.class}" name="classname"/>**

**<attribute default="${debug.modulepath}" name="modulepath"/>**

**<attribute default="${debug.classpath}" name="classpath"/>**

**<element name="customizeDebuggee" optional="true"/>**

**<sequential>**

**<j2seproject1:java classname="@{classname}" classpath="@{classpath}" modulename="@{modulename}" modulepath="@{modulepath}">**

**<customize>**

**<jvmarg value="-agentlib:jdwp=transport=${debug-transport},address=${jpda.address}"/>**

**<customizeDebuggee/>**

**</customize>**

**</j2seproject1:java>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-source-module-properties" if="named.module.internal" name="-init-macrodef-java-with-module">**

**<macrodef name="java" uri="http://www.netbeans.org/ns/j2se-project/1">**

**<attribute default="${module.name}" name="modulename"/>**

**<attribute default="${main.class}" name="classname"/>**

**<attribute default="${run.modulepath}" name="modulepath"/>**

**<attribute default="${run.upgrademodulepath}" name="upgrademodulepath"/>**

**<attribute default="${run.classpath}" name="classpath"/>**

**<attribute default="jvm" name="jvm"/>**

**<element name="customize" optional="true"/>**

**<sequential>**

**<java classname="@{classname}" dir="${work.dir}" failonerror="${java.failonerror}" fork="true" module="@{modulename}">**

**<classpath>**

**<path path="@{classpath}"/>**

**</classpath>**

**<modulepath>**

**<pathelement path="@{modulepath}"/>**

**<pathelement location="${module.build.classes.dir}"/>**

**</modulepath>**

**<upgrademodulepath>**

**<path path="@{upgrademodulepath}"/>**

**</upgrademodulepath>**

**<jvmarg value="-Dfile.encoding=${runtime.encoding}"/>**

**<redirector errorencoding="${runtime.encoding}" inputencoding="${runtime.encoding}" outputencoding="${runtime.encoding}"/>**

**<jvmarg line="${run.jvmargs}"/>**

**<jvmarg line="${run.jvmargs.ide}"/>**

**<syspropertyset>**

**<propertyref prefix="run-sys-prop."/>**

**<mapper from="run-sys-prop.\*" to="\*" type="glob"/>**

**</syspropertyset>**

**<customize/>**

**</java>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-source-module-properties" if="unnamed.module.internal" name="-init-macrodef-java-with-unnamed-module">**

**<macrodef name="java" uri="http://www.netbeans.org/ns/j2se-project/1">**

**<attribute default="" name="modulename"/>**

**<attribute default="${main.class}" name="classname"/>**

**<attribute default="${run.modulepath}" name="modulepath"/>**

**<attribute default="${run.upgrademodulepath}" name="upgrademodulepath"/>**

**<attribute default="${run.classpath}" name="classpath"/>**

**<attribute default="jvm" name="jvm"/>**

**<element name="customize" optional="true"/>**

**<sequential>**

**<java classname="@{classname}" dir="${work.dir}" failonerror="${java.failonerror}" fork="true">**

**<classpath>**

**<path path="@{classpath}"/>**

**</classpath>**

**<modulepath>**

**<path path="@{modulepath}"/>**

**</modulepath>**

**<upgrademodulepath>**

**<path path="@{upgrademodulepath}"/>**

**</upgrademodulepath>**

**<jvmarg value="-Dfile.encoding=${runtime.encoding}"/>**

**<redirector errorencoding="${runtime.encoding}" inputencoding="${runtime.encoding}" outputencoding="${runtime.encoding}"/>**

**<jvmarg line="${run.jvmargs}"/>**

**<jvmarg line="${run.jvmargs.ide}"/>**

**<syspropertyset>**

**<propertyref prefix="run-sys-prop."/>**

**<mapper from="run-sys-prop.\*" to="\*" type="glob"/>**

**</syspropertyset>**

**<customize/>**

**</java>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-source-module-properties" name="-init-macrodef-java-without-module" unless="modules.supported.internal">**

**<macrodef name="java" uri="http://www.netbeans.org/ns/j2se-project/1">**

**<attribute default="" name="modulename"/>**

**<attribute default="${main.class}" name="classname"/>**

**<attribute default="" name="modulepath"/>**

**<attribute default="${run.classpath}" name="classpath"/>**

**<attribute default="jvm" name="jvm"/>**

**<element name="customize" optional="true"/>**

**<sequential>**

**<java classname="@{classname}" dir="${work.dir}" failonerror="${java.failonerror}" fork="true">**

**<jvmarg line="${endorsed.classpath.cmd.line.arg}"/>**

**<jvmarg value="-Dfile.encoding=${runtime.encoding}"/>**

**<redirector errorencoding="${runtime.encoding}" inputencoding="${runtime.encoding}" outputencoding="${runtime.encoding}"/>**

**<jvmarg line="${run.jvmargs}"/>**

**<jvmarg line="${run.jvmargs.ide}"/>**

**<classpath>**

**<path path="@{classpath}"/>**

**</classpath>**

**<syspropertyset>**

**<propertyref prefix="run-sys-prop."/>**

**<mapper from="run-sys-prop.\*" to="\*" type="glob"/>**

**</syspropertyset>**

**<customize/>**

**</java>**

**</sequential>**

**</macrodef>**

**</target>**

**<target depends="-init-macrodef-java-with-module, -init-macrodef-java-with-unnamed-module, -init-macrodef-java-without-module" name="-init-macrodef-java"/>**

**<target name="-init-macrodef-copylibs">**

**<macrodef name="copylibs" uri="http://www.netbeans.org/ns/j2se-project/3">**

**<attribute default="${manifest.file}" name="manifest"/>**

**<element name="customize" optional="true"/>**

**<sequential>**

**<property location="${build.classes.dir}" name="build.classes.dir.resolved"/>**

**<pathconvert property="run.classpath.without.build.classes.dir">**

**<path path="${run.classpath}"/>**

**<map from="${build.classes.dir.resolved}" to=""/>**

**</pathconvert>**

**<pathconvert pathsep=" " property="jar.classpath">**

**<path path="${run.classpath.without.build.classes.dir}"/>**

**<chainedmapper>**

**<flattenmapper/>**

**<filtermapper>**

**<replacestring from=" " to="%20"/>**

**</filtermapper>**

**<globmapper from="\*" to="lib/\*"/>**

**</chainedmapper>**

**</pathconvert>**

**<taskdef classname="org.netbeans.modules.java.j2seproject.copylibstask.CopyLibs" classpath="${libs.CopyLibs.classpath}" name="copylibs"/>**

**<copylibs compress="${jar.compress}" excludeFromCopy="${copylibs.excludes}" index="${jar.index}" indexMetaInf="${jar.index.metainf}" jarfile="${dist.jar}" manifest="@{manifest}" manifestencoding="UTF-8" rebase="${copylibs.rebase}" runtimeclasspath="${run.classpath.without.build.classes.dir}">**

**<fileset dir="${build.classes.dir}" excludes="${dist.archive.excludes}"/>**

**<manifest>**

**<attribute name="Class-Path" value="${jar.classpath}"/>**

**<customize/>**

**</manifest>**

**</copylibs>**

**</sequential>**

**</macrodef>**

**</target>**

**<target name="-init-presetdef-jar">**

**<presetdef name="jar" uri="http://www.netbeans.org/ns/j2se-project/1">**

**<jar compress="${jar.compress}" index="${jar.index}" jarfile="${dist.jar}" manifestencoding="UTF-8">**

**<j2seproject1:fileset dir="${build.classes.dir}" excludes="${dist.archive.excludes}"/>**

**</jar>**

**</presetdef>**

**</target>**

**<target name="-init-ap-cmdline-properties">**

**<property name="annotation.processing.enabled" value="true"/>**

**<property name="annotation.processing.processors.list" value=""/>**

**<property name="annotation.processing.processor.options" value=""/>**

**<property name="annotation.processing.run.all.processors" value="true"/>**

**<property name="javac.processorpath" value="${javac.classpath}"/>**

**<property name="javac.test.processorpath" value="${javac.test.classpath}"/>**

**<condition property="ap.supported.internal" value="true">**

**<not>**

**<matches pattern="1\.[0-5](\..\*)?" string="${javac.source}"/>**

**</not>**

**</condition>**

**</target>**

**<target depends="-init-ap-cmdline-properties" if="ap.supported.internal" name="-init-ap-cmdline-supported">**

**<condition else="" property="ap.processors.internal" value="-processor ${annotation.processing.processors.list}">**

**<isfalse value="${annotation.processing.run.all.processors}"/>**

**</condition>**

**<condition else="" property="ap.proc.none.internal" value="-proc:none">**

**<isfalse value="${annotation.processing.enabled}"/>**

**</condition>**

**</target>**

**<target depends="-init-ap-cmdline-properties,-init-ap-cmdline-supported" name="-init-ap-cmdline">**

**<property name="ap.cmd.line.internal" value=""/>**

**</target>**

**<target depends="-pre-init,-init-private,-init-user,-init-project,-do-init,-post-init,-init-check,-init-macrodef-property,-init-macrodef-javac,-init-macrodef-test,-init-macrodef-test-debug,-init-macrodef-nbjpda,-init-macrodef-debug,-init-macrodef-java,-init-presetdef-jar,-init-ap-cmdline" name="init"/>**

**<!--**

**===================**

**COMPILATION SECTION**

**===================**

**-->**

**<target name="-deps-jar-init" unless="built-jar.properties">**

**<property location="${build.dir}/built-jar.properties" name="built-jar.properties"/>**

**<delete file="${built-jar.properties}" quiet="true"/>**

**</target>**

**<target if="already.built.jar.${basedir}" name="-warn-already-built-jar">**

**<echo level="warn" message="Cycle detected: Banking Management System was already built"/>**

**</target>**

**<target depends="init,-deps-jar-init" name="deps-jar" unless="no.deps">**

**<mkdir dir="${build.dir}"/>**

**<touch file="${built-jar.properties}" verbose="false"/>**

**<property file="${built-jar.properties}" prefix="already.built.jar."/>**

**<antcall target="-warn-already-built-jar"/>**

**<propertyfile file="${built-jar.properties}">**

**<entry key="${basedir}" value=""/>**

**</propertyfile>**

**</target>**

**<target depends="init,-check-automatic-build,-clean-after-automatic-build" name="-verify-automatic-build"/>**

**<target depends="init" name="-check-automatic-build">**

**<available file="${build.classes.dir}/.netbeans\_automatic\_build" property="netbeans.automatic.build"/>**

**</target>**

**<target depends="init" if="netbeans.automatic.build" name="-clean-after-automatic-build">**

**<antcall target="clean">**

**<param name="no.dependencies" value="true"/>**

**</antcall>**

**</target>**

**<target depends="init,deps-jar" name="-pre-pre-compile">**

**<mkdir dir="${build.classes.dir}"/>**

**</target>**

**<target name="-pre-compile">**

**<!-- Empty placeholder for easier customization. -->**

**<!-- You can override this target in the ../build.xml file. -->**

**</target>**

**<target if="do.depend.true" name="-compile-depend">**

**<pathconvert property="build.generated.subdirs">**

**<dirset dir="${build.generated.sources.dir}" erroronmissingdir="false">**

**<include name="\*"/>**

**</dirset>**

**</pathconvert>**

**<j2seproject3:depend srcdir="${src.dir}:${build.generated.subdirs}"/>**

**</target>**

**<target depends="init,deps-jar,-pre-pre-compile,-pre-compile, -copy-persistence-xml,-compile-depend" if="have.sources" name="-do-compile">**

**<j2seproject3:javac gensrcdir="${build.generated.sources.dir}"/>**

**<copy todir="${build.classes.dir}">**

**<fileset dir="${src.dir}" excludes="${build.classes.excludes},${excludes}" includes="${includes}"/>**

**</copy>**

**</target>**

**<target if="has.persistence.xml" name="-copy-persistence-xml">**

**<mkdir dir="${build.classes.dir}/META-INF"/>**

**<copy todir="${build.classes.dir}/META-INF">**

**<fileset dir="${meta.inf.dir}" includes="persistence.xml orm.xml"/>**

**</copy>**

**</target>**

**<target name="-post-compile">**

**<!-- Empty placeholder for easier customization. -->**

**<!-- You can override this target in the ../build.xml file. -->**

**</target>**

**<target depends="init,deps-jar,-verify-automatic-build,-pre-pre-compile,-pre-compile,-do-compile,-post-compile" description="Compile project." name="compile"/>**

**<target name="-pre-compile-single">**

**<!-- Empty placeholder for easier customization. -->**

**<!-- You can override this target in the ../build.xml file. -->**

**</target>**

**<target depends="init,deps-jar,-pre-pre-compile" name="-do-compile-single">**

**<fail unless="javac.includes">Must select some files in the IDE or set javac.includes</fail>**

**<j2seproject3:force-recompile/>**

**<j2seproject3:javac excludes="" gensrcdir="${build.generated.sources.dir}" includes="${javac.includes}, module-info.java" sourcepath="${src.dir}"/>**

**</target>**

**<target name="-post-compile-single">**

**<!-- Empty placeholder for easier customization. -->**

**<!-- You can override this target in the ../build.xml file. -->**

**</target>**

**<target depends="init,deps-jar,-verify-automatic-build,-pre-pre-compile,-pre-compile-single,-do-compile-single,-post-compile-single" name="compile-single"/>**

**<!--**

**====================**

**JAR BUILDING SECTION**

**====================**

**-->**

**<target depends="init" name="-pre-pre-jar">**

**<dirname file="${dist.jar}" property="dist.jar.dir"/>**

**<mkdir dir="${dist.jar.dir}"/>**

**</target>**

**<target name="-pre-jar">**

**<!-- Empty placeholder for easier customization. -->**

**<!-- You can override this target in the ../build.xml file. -->**

**</target>**

**<target depends="init,compile" name="-check-module-main-class">**

**<pathconvert property="main.class.file">**

**<string value="${main.class}"/>**

**<unpackagemapper from="\*" to="\*.class"/>**

**</pathconvert>**

**<condition property="do.module.main.class">**

**<and>**

**<isset property="main.class.available"/>**

**<available file="${build.classes.dir}/module-info.class"/>**

**<available file="${build.classes.dir}/${main.class.file}"/>**

**<isset property="libs.CopyLibs.classpath"/>**

**<available classname="org.netbeans.modules.java.j2seproject.moduletask.ModuleMainClass" classpath="${libs.CopyLibs.classpath}"/>**

**</and>**

**</condition>**

**</target>**

**<target depends="-check-module-main-class" if="do.module.main.class" name="-set-module-main-class">**

**<taskdef classname="org.netbeans.modules.java.j2seproject.moduletask.ModuleMainClass" classpath="${libs.CopyLibs.classpath}" name="modulemainclass"/>**

**<modulemainclass failonerror="false" mainclass="${main.class}" moduleinfo="${build.classes.dir}/module-info.class"/>**

**</target>**

**<target depends="init" if="do.archive" name="-do-jar-create-manifest" unless="manifest.available">**

**<tempfile deleteonexit="true" destdir="${build.dir}" property="tmp.manifest.file"/>**

**<touch file="${tmp.manifest.file}" verbose="false"/>**

**</target>**

**<target depends="init" if="do.archive+manifest.available" name="-do-jar-copy-manifest">**

**<tempfile deleteonexit="true" destdir="${build.dir}" property="tmp.manifest.file"/>**

**<copy encoding="${manifest.encoding}" file="${manifest.file}" outputencoding="UTF-8" tofile="${tmp.manifest.file}"/>**

**</target>**

**<target depends="init,-do-jar-create-manifest,-do-jar-copy-manifest" if="do.archive+main.class.available" name="-do-jar-set-mainclass">**

**<manifest encoding="UTF-8" file="${tmp.manifest.file}" mode="update">**

**<attribute name="Main-Class" value="${main.class}"/>**

**</manifest>**

**</target>**

**<target depends="init,-do-jar-create-manifest,-do-jar-copy-manifest" if="do.archive+profile.available" name="-do-jar-set-profile">**

**<manifest encoding="UTF-8" file="${tmp.manifest.file}" mode="update">**

**<attribute name="Profile" value="${javac.profile}"/>**

**</manifest>**

**</target>**

**<target depends="init,-do-jar-create-manifest,-do-jar-copy-manifest" if="do.archive+splashscreen.available" name="-do-jar-set-splashscreen">**

**<basename file="${application.splash}" property="splashscreen.basename"/>**

**<mkdir dir="${build.classes.dir}/META-INF"/>**

**<copy failonerror="false" file="${application.splash}" todir="${build.classes.dir}/META-INF"/>**

**<manifest encoding="UTF-8" file="${tmp.manifest.file}" mode="update">**

**<attribute name="SplashScreen-Image" value="META-INF/${splashscreen.basename}"/>**

**</manifest>**

**</target>**

**<target depends="init,compile" name="-check-do-mkdist">**

**<condition property="do.mkdist">**

**<and>**

**<isset property="do.archive"/>**

**<isset property="libs.CopyLibs.classpath"/>**

**<not>**

**<istrue value="${mkdist.disabled}"/>**

**</not>**

**<not>**

**<available file="${build.classes.dir}/module-info.class"/>**

**</not>**

**</and>**

**</condition>**

**</target>**

**<target depends="init,-init-macrodef-copylibs,compile,-pre-pre-jar,-pre-jar,-do-jar-create-manifest,-do-jar-copy-manifest,-do-jar-set-mainclass,-do-jar-set-profile,-do-jar-set-splashscreen,-check-do-mkdist" if="do.mkdist" name="-do-jar-copylibs">**

**<j2seproject3:copylibs manifest="${tmp.manifest.file}"/>**

**<echo level="info">To run this application from the command line without Ant, try:</echo>**

**<property location="${dist.jar}" name="dist.jar.resolved"/>**

**<echo level="info">java -jar "${dist.jar.resolved}"</echo>**

**</target>**

**<target depends="init,compile,-pre-pre-jar,-pre-jar,-do-jar-create-manifest,-do-jar-copy-manifest,-do-jar-set-mainclass,-do-jar-set-profile,-do-jar-set-splashscreen,-check-do-mkdist" if="do.archive" name="-do-jar-jar" unless="do.mkdist">**

**<j2seproject1:jar manifest="${tmp.manifest.file}"/>**

**<property location="${build.classes.dir}" name="build.classes.dir.resolved"/>**

**<property location="${dist.jar}" name="dist.jar.resolved"/>**

**<condition else="${dist.jar.resolved}" property="jar.usage.message.class.path.replacement" value="">**

**<isset property="named.module.internal"/>**

**</condition>**

**<pathconvert property="run.classpath.with.dist.jar">**

**<path path="${run.classpath}"/>**

**<map from="${build.classes.dir.resolved}" to="${jar.usage.message.class.path.replacement}"/>**

**</pathconvert>**

**<pathconvert property="run.modulepath.with.dist.jar">**

**<path location="${dist.jar.resolved}"/>**

**<path path="${run.modulepath}"/>**

**<map from="${build.classes.dir.resolved}" to="${dist.jar.resolved}"/>**

**</pathconvert>**

**<condition else="${run.modulepath}" property="jar.usage.message.run.modulepath.with.dist.jar" value="${run.modulepath.with.dist.jar}">**

**<isset property="named.module.internal"/>**

**</condition>**

**<condition else="" property="jar.usage.message.module.path" value=" -p ${jar.usage.message.run.modulepath.with.dist.jar}">**

**<and>**

**<isset property="modules.supported.internal"/>**

**<length length="0" string="${jar.usage.message.run.modulepath.with.dist.jar}" when="greater"/>**

**</and>**

**</condition>**

**<condition else="" property="jar.usage.message.class.path" value=" -cp ${run.classpath.with.dist.jar}">**

**<length length="0" string="${run.classpath.with.dist.jar}" when="greater"/>**

**</condition>**

**<condition else="/${main.class}" property="jar.usage.message.main.class.class.selector" value="">**

**<isset property="do.module.main.class"/>**

**</condition>**

**<condition else=" ${main.class}" property="jar.usage.message.main.class" value=" -m ${module.name}${jar.usage.message.main.class.class.selector}">**

**<isset property="named.module.internal"/>**

**</condition>**

**<condition else="" property="jar.usage.message" value="To run this application from the command line without Ant, try:${line.separator}${platform.java}${jar.usage.message.module.path}${jar.usage.message.class.path}${jar.usage.message.main.class}">**

**<isset property="main.class.available"/>**

**</condition>**

**<condition else="debug" property="jar.usage.level" value="info">**

**<isset property="main.class.available"/>**

**</condition>**

**<echo level="${jar.usage.level}" message="${jar.usage.message}"/>**

**</target>**

**<target depends="-do-jar-copylibs" if="do.archive" name="-do-jar-delete-manifest">**

**<delete>**

**<fileset file="${tmp.manifest.file}"/>**

**</delete>**

**</target>**

**<target depends="init,compile,-pre-pre-jar,-pre-jar,-do-jar-create-manifest,-do-jar-copy-manifest,-do-jar-set-mainclass,-do-jar-set-profile,-do-jar-set-splashscreen,-do-jar-jar,-do-jar-delete-manifest" name="-do-jar-without-libraries"/>**

**<target depends="init,compile,-pre-pre-jar,-pre-jar,-do-jar-create-manifest,-do-jar-copy-manifest,-do-jar-set-mainclass,-do-jar-set-profile,-do-jar-set-splashscreen,-do-jar-copylibs,-do-jar-delete-manifest" name="-do-jar-with-libraries"/>**

**<target name="-post-jar">**

**<!-- Empty placeholder for easier customization. -->**

**<!-- You can override this target in the ../build.xml file. -->**

**</target>**

**<target depends="init,compile,-pre-jar,-set-module-main-class,-do-jar-without-libraries,-do-jar-with-libraries,-post-jar" name="-do-jar"/>**

**<target depends="init,compile,-pre-jar,-do-jar,-post-jar,deploy" description="Build JAR." name="jar"/>**

**<!--**

**=================**

**DEPLOY SECTION**

**=================**

**-->**

**<target name="-pre-deploy">**

**<!-- Empty placeholder for easier customization. -->**

**<!-- You can override this target in the ../build.xml file. -->**

**</target>**

**<target depends="init" name="-check-jlink">**

**<condition property="do.jlink.internal">**

**<and>**

**<istrue value="${do.jlink}"/>**

**<isset property="do.archive"/>**

**<isset property="named.module.internal"/>**

**</and>**

**</condition>**

**</target>**

**<target depends="init,-do-jar,-post-jar,-pre-deploy,-check-jlink" if="do.jlink.internal" name="-do-deploy">**

**<delete dir="${dist.jlink.dir}" failonerror="false" quiet="true"/>**

**<property name="jlink.launcher.name" value="${application.title}"/>**

**<condition else="${module.name}" property="jlink.add.modules" value="${module.name},${jlink.additionalmodules}">**

**<and>**

**<isset property="jlink.additionalmodules"/>**

**<length length="0" string="${jlink.additionalmodules}" when="greater"/>**

**</and>**

**</condition>**

**<condition property="jlink.do.strip.internal">**

**<and>**

**<isset property="jlink.strip"/>**

**<istrue value="${jlink.strip}"/>**

**</and>**

**</condition>**

**<condition property="jlink.do.additionalparam.internal">**

**<and>**

**<isset property="jlink.additionalparam"/>**

**<length length="0" string="${jlink.additionalparam}" when="greater"/>**

**</and>**

**</condition>**

**<condition property="jlink.do.launcher.internal">**

**<and>**

**<istrue value="${jlink.launcher}"/>**

**<isset property="main.class.available"/>**

**</and>**

**</condition>**

**<property name="platform.jlink" value="${jdk.home}/bin/jlink"/>**

**<property name="jlink.systemmodules.internal" value="${jdk.home}/jmods"/>**

**<exec executable="${platform.jlink}">**

**<arg value="--module-path"/>**

**<arg path="${jlink.systemmodules.internal}:${run.modulepath}:${dist.jar}"/>**

**<arg value="--add-modules"/>**

**<arg value="${jlink.add.modules}"/>**

**<arg if:set="jlink.do.strip.internal" value="--strip-debug"/>**

**<arg if:set="jlink.do.launcher.internal" value="--launcher"/>**

**<arg if:set="jlink.do.launcher.internal" value="${jlink.launcher.name}=${module.name}/${main.class}"/>**

**<arg if:set="jlink.do.additionalparam.internal" line="${jlink.additionalparam}"/>**

**<arg value="--output"/>**

**<arg value="${dist.jlink.output}"/>**

**</exec>**

**</target>**

**<target name="-post-deploy">**

**<!-- Empty placeholder for easier customization. -->**

**<!-- You can override this target in the ../build.xml file. -->**

**</target>**

**<target depends="-do-jar,-post-jar,-pre-deploy,-do-deploy,-post-deploy" name="deploy"/>**

**<!--**

**=================**

**EXECUTION SECTION**

**=================**

**-->**

**<target depends="init,compile" description="Run a main class." name="run">**

**<j2seproject1:java>**

**<customize>**

**<arg line="${application.args}"/>**

**</customize>**

**</j2seproject1:java>**

**</target>**

**<target name="-do-not-recompile">**

**<property name="javac.includes.binary" value=""/>**

**</target>**

**<target depends="init,compile-single" name="run-single">**

**<fail unless="run.class">Must select one file in the IDE or set run.class</fail>**

**<j2seproject1:java classname="${run.class}"/>**

**</target>**

**<target depends="init,compile-test-single" name="run-test-with-main">**

**<fail unless="run.class">Must select one file in the IDE or set run.class</fail>**

**<j2seproject1:java classname="${run.class}" classpath="${run.test.classpath}"/>**

**</target>**

**<!--**

**=================**

**DEBUGGING SECTION**

**=================**

**-->**

**<target depends="init" if="netbeans.home" name="-debug-start-debugger">**

**<j2seproject1:nbjpdastart name="${debug.class}"/>**

**</target>**

**<target depends="init" if="netbeans.home" name="-debug-start-debugger-main-test">**

**<j2seproject1:nbjpdastart classpath="${debug.test.classpath}" name="${debug.class}"/>**

**</target>**

**<target depends="init,compile" name="-debug-start-debuggee">**

**<j2seproject3:debug>**

**<customizeDebuggee>**

**<arg line="${application.args}"/>**

**</customizeDebuggee>**

**</j2seproject3:debug>**

**</target>**

**<target depends="init,compile,-debug-start-debugger,-debug-start-debuggee" description="Debug project in IDE." if="netbeans.home" name="debug"/>**

**<target depends="init" if="netbeans.home" name="-debug-start-debugger-stepinto">**

**<j2seproject1:nbjpdastart stopclassname="${main.class}"/>**

**</target>**

**<target depends="init,compile,-debug-start-debugger-stepinto,-debug-start-debuggee" if="netbeans.home" name="debug-stepinto"/>**

**<target depends="init,compile-single" if="netbeans.home" name="-debug-start-debuggee-single">**

**<fail unless="debug.class">Must select one file in the IDE or set debug.class</fail>**

**<j2seproject3:debug classname="${debug.class}"/>**

**</target>**

**<target depends="init,compile-single,-debug-start-debugger,-debug-start-debuggee-single" if="netbeans.home" name="debug-single"/>**

**<target depends="init,compile-test-single" if="netbeans.home" name="-debug-start-debuggee-main-test">**

**<fail unless="debug.class">Must select one file in the IDE or set debug.class</fail>**

**<j2seproject3:debug classname="${debug.class}" classpath="${debug.test.classpath}"/>**

**</target>**

**<target depends="init,compile-test-single,-debug-start-debugger-main-test,-debug-start-debuggee-main-test" if="netbeans.home" name="debug-test-with-main"/>**

**<target depends="init" name="-pre-debug-fix">**

**<fail unless="fix.includes">Must set fix.includes</fail>**

**<property name="javac.includes" value="${fix.includes}.java"/>**

**</target>**

**<target depends="init,-pre-debug-fix,compile-single" if="netbeans.home" name="-do-debug-fix">**

**<j2seproject1:nbjpdareload/>**

**</target>**

**<target depends="init,-pre-debug-fix,-do-debug-fix" if="netbeans.home" name="debug-fix"/>**

**<!--**

**=================**

**PROFILING SECTION**

**=================**

**-->**

**<!--**

**pre NB7.2 profiler integration**

**-->**

**<target depends="profile-init,compile" description="Profile a project in the IDE." if="profiler.info.jvmargs.agent" name="-profile-pre72">**

**<fail unless="netbeans.home">This target only works when run from inside the NetBeans IDE.</fail>**

**<nbprofiledirect>**

**<classpath>**

**<path path="${run.classpath}"/>**

**</classpath>**

**</nbprofiledirect>**

**<profile/>**

**</target>**

**<target depends="profile-init,compile-single" description="Profile a selected class in the IDE." if="profiler.info.jvmargs.agent" name="-profile-single-pre72">**

**<fail unless="profile.class">Must select one file in the IDE or set profile.class</fail>**

**<fail unless="netbeans.home">This target only works when run from inside the NetBeans IDE.</fail>**

**<nbprofiledirect>**

**<classpath>**

**<path path="${run.classpath}"/>**

**</classpath>**

**</nbprofiledirect>**

**<profile classname="${profile.class}"/>**

**</target>**

**<target depends="profile-init,compile-single" if="profiler.info.jvmargs.agent" name="-profile-applet-pre72">**

**<fail unless="netbeans.home">This target only works when run from inside the NetBeans IDE.</fail>**

**<nbprofiledirect>**

**<classpath>**

**<path path="${run.classpath}"/>**

**</classpath>**

**</nbprofiledirect>**

**<profile classname="sun.applet.AppletViewer">**

**<customize>**

**<arg value="${applet.url}"/>**

**</customize>**

**</profile>**

**</target>**

**<target depends="-init-macrodef-junit,profile-init,compile-test-single" if="profiler.info.jvmargs.agent" name="-profile-test-single-pre72">**

**<fail unless="netbeans.home">This target only works when run from inside the NetBeans IDE.</fail>**

**<nbprofiledirect>**

**<classpath>**

**<path path="${run.test.classpath}"/>**

**</classpath>**

**</nbprofiledirect>**

**<j2seproject3:junit excludes="${excludes}" includes="${includes}" testincludes="${profile.class}" testmethods="">**

**<customize>**

**<jvmarg value="-agentlib:jdwp=transport=${debug-transport},address=${jpda.address}"/>**

**<env key="${profiler.info.pathvar}" path="${profiler.info.agentpath}:${profiler.current.path}"/>**

**<jvmarg value="${profiler.info.jvmargs.agent}"/>**

**<jvmarg line="${profiler.info.jvmargs}"/>**

**<classpath>**

**<path path="${run.test.classpath}"/>**

**</classpath>**

**</customize>**

**</j2seproject3:junit>**

**</target>**

**<!--**

**end of pre NB72 profiling section**

**-->**

**<target if="netbeans.home" name="-profile-check">**

**<condition property="profiler.configured">**

**<or>**

**<contains casesensitive="true" string="${run.jvmargs.ide}" substring="-agentpath:"/>**

**<contains casesensitive="true" string="${run.jvmargs.ide}" substring="-javaagent:"/>**

**</or>**

**</condition>**

**</target>**

**<target depends="-profile-check,-profile-pre72" description="Profile a project in the IDE." if="profiler.configured" name="profile" unless="profiler.info.jvmargs.agent">**

**<startprofiler/>**

**<antcall target="run"/>**

**</target>**

**<target depends="-profile-check,-profile-single-pre72" description="Profile a selected class in the IDE." if="profiler.configured" name="profile-single" unless="profiler.info.jvmargs.agent">**

**<fail unless="run.class">Must select one file in the IDE or set run.class</fail>**

**<startprofiler/>**

**<antcall target="run-single"/>**

**</target>**

**<target depends="-profile-test-single-pre72" description="Profile a selected test in the IDE." name="profile-test-single"/>**

**<target depends="-profile-check" description="Profile a selected test in the IDE." if="profiler.configured" name="profile-test" unless="profiler.info.jvmargs">**

**<fail unless="test.includes">Must select some files in the IDE or set test.includes</fail>**

**<startprofiler/>**

**<antcall target="test-single"/>**

**</target>**

**<target depends="-profile-check" description="Profile a selected class in the IDE." if="profiler.configured" name="profile-test-with-main">**

**<fail unless="run.class">Must select one file in the IDE or set run.class</fail>**

**<startprofiler/>**

**<antcall target="run-test-with-main"/>**

**</target>**

**<target depends="-profile-check,-profile-applet-pre72" if="profiler.configured" name="profile-applet" unless="profiler.info.jvmargs.agent">**

**<fail unless="applet.url">Must select one file in the IDE or set applet.url</fail>**

**<startprofiler/>**

**<antcall target="run-applet"/>**

**</target>**

**<!--**

**===============**

**JAVADOC SECTION**

**===============**

**-->**

**<target depends="init" if="have.sources" name="-javadoc-build">**

**<mkdir dir="${dist.javadoc.dir}"/>**

**<condition else="" property="javadoc.endorsed.classpath.cmd.line.arg" value="-J${endorsed.classpath.cmd.line.arg}">**

**<and>**

**<isset property="endorsed.classpath.cmd.line.arg"/>**

**<not>**

**<equals arg1="${endorsed.classpath.cmd.line.arg}" arg2=""/>**

**</not>**

**</and>**

**</condition>**

**<condition else="" property="bug5101868workaround" value="\*.java">**

**<matches pattern="1\.[56](\..\*)?" string="${java.version}"/>**

**</condition>**

**<condition else="" property="javadoc.html5.cmd.line.arg" value="-html5">**

**<and>**

**<isset property="javadoc.html5"/>**

**<available file="${jdk.home}${file.separator}lib${file.separator}jrt-fs.jar"/>**

**</and>**

**</condition>**

**<javadoc additionalparam="-J-Dfile.encoding=${file.encoding} ${javadoc.additionalparam}" author="${javadoc.author}" charset="UTF-8" destdir="${dist.javadoc.dir}" docencoding="UTF-8" encoding="${javadoc.encoding.used}" failonerror="true" noindex="${javadoc.noindex}" nonavbar="${javadoc.nonavbar}" notree="${javadoc.notree}" private="${javadoc.private}" source="${javac.source}" splitindex="${javadoc.splitindex}" use="${javadoc.use}" useexternalfile="true" version="${javadoc.version}" windowtitle="${javadoc.windowtitle}">**

**<classpath>**

**<path path="${javac.classpath}"/>**

**</classpath>**

**<fileset dir="${src.dir}" excludes="${bug5101868workaround},${excludes}" includes="${includes}">**

**<filename name="\*\*/\*.java"/>**

**</fileset>**

**<fileset dir="${build.generated.sources.dir}" erroronmissingdir="false">**

**<include name="\*\*/\*.java"/>**

**<exclude name="\*.java"/>**

**</fileset>**

**<arg line="${javadoc.endorsed.classpath.cmd.line.arg}"/>**

**<arg line="${javadoc.html5.cmd.line.arg}"/>**

**</javadoc>**

**<copy todir="${dist.javadoc.dir}">**

**<fileset dir="${src.dir}" excludes="${excludes}" includes="${includes}">**

**<filename name="\*\*/doc-files/\*\*"/>**

**</fileset>**

**<fileset dir="${build.generated.sources.dir}" erroronmissingdir="false">**

**<include name="\*\*/doc-files/\*\*"/>**

**</fileset>**

**</copy>**

**</target>**

**<target depends="init,-javadoc-build" if="netbeans.home" name="-javadoc-browse" unless="no.javadoc.preview">**

**<nbbrowse file="${dist.javadoc.dir}/index.html"/>**

**</target>**

**<target depends="init,-javadoc-build,-javadoc-browse" description="Build Javadoc." name="javadoc"/>**

**<!--**

**=========================**

**TEST COMPILATION SECTION**

**=========================**

**-->**

**<target depends="init,compile" if="have.tests" name="-pre-pre-compile-test">**

**<mkdir dir="${build.test.classes.dir}"/>**

**</target>**

**<target name="-pre-compile-test">**

**<!-- Empty placeholder for easier customization. -->**

**<!-- You can override this target in the ../build.xml file. -->**

**</target>**

**<target depends="-init-source-module-properties" if="named.module.internal" name="-init-test-javac-module-properties-with-module">**

**<j2seproject3:modulename property="test.module.name" sourcepath="${test.src.dir}"/>**

**<condition else="${empty.dir}" property="javac.test.sourcepath" value="${test.src.dir}">**

**<and>**

**<isset property="test.module.name"/>**

**<length length="0" string="${test.module.name}" when="greater"/>**

**</and>**

**</condition>**

**<condition else="--patch-module ${module.name}=${test.src.dir} --add-reads ${module.name}=ALL-UNNAMED" property="javac.test.compilerargs" value="--add-reads ${test.module.name}=ALL-UNNAMED">**

**<and>**

**<isset property="test.module.name"/>**

**<length length="0" string="${test.module.name}" when="greater"/>**

**</and>**

**</condition>**

**</target>**

**<target depends="-init-source-module-properties" if="named.module.internal" name="-init-test-run-module-properties">**

**<condition else="${module.name}" property="run.test.addexport.source.module.internal" value="${test.module.name}">**

**<and>**

**<isset property="test.module.name"/>**

**<length length="0" string="${test.module.name}" when="greater"/>**

**</and>**

**</condition>**

**<fileset dir="${build.test.classes.dir}" id="run.test.packages.internal" includes="\*\*/\*.class"/>**

**<property location="${build.test.classes.dir}" name="build.test.classes.dir.abs.internal"/>**

**<pathconvert pathsep=" " property="run.test.addexports.internal" refid="run.test.packages.internal">**

**<chainedmapper>**

**<regexpmapper from="^(.\*)\Q${file.separator}\E.\*\.class$$" to="\1"/>**

**<filtermapper>**

**<uniqfilter/>**

**<replacestring from="${build.test.classes.dir.abs.internal}" to=""/>**

**</filtermapper>**

**<cutdirsmapper dirs="1"/>**

**<packagemapper from="\*" to="--add-exports ${run.test.addexport.source.module.internal}/\*=ALL-UNNAMED"/>**

**</chainedmapper>**

**</pathconvert>**

**<condition else="--patch-module ${module.name}=${build.test.classes.dir} --add-modules ${module.name} --add-reads ${module.name}=ALL-UNNAMED ${run.test.addexports.internal}" property="run.test.jvmargs" value="--add-modules ${test.module.name} --add-reads ${test.module.name}=ALL-UNNAMED ${run.test.addexports.internal}">**

**<and>**

**<isset property="test.module.name"/>**

**<length length="0" string="${test.module.name}" when="greater"/>**

**</and>**

**</condition>**

**</target>**

**<target depends="-init-source-module-properties" name="-init-test-module-properties-without-module" unless="named.module.internal">**

**<property name="javac.test.sourcepath" value="${empty.dir}"/>**

**<property name="javac.test.compilerargs" value=""/>**

**<property name="run.test.jvmargs" value=""/>**

**</target>**

**<target depends="-init-test-javac-module-properties-with-module,-init-test-module-properties-without-module" name="-init-test-module-properties"/>**

**<target if="do.depend.true" name="-compile-test-depend">**

**<j2seproject3:depend classpath="${javac.test.classpath}" destdir="${build.test.classes.dir}" srcdir="${test.src.dir}"/>**

**</target>**

**<target depends="init,deps-jar,compile,-init-test-module-properties,-pre-pre-compile-test,-pre-compile-test,-compile-test-depend" if="have.tests" name="-do-compile-test">**

**<j2seproject3:javac apgeneratedsrcdir="${build.test.classes.dir}" classpath="${javac.test.classpath}" debug="true" destdir="${build.test.classes.dir}" modulepath="${javac.test.modulepath}" processorpath="${javac.test.processorpath}" sourcepath="${javac.test.sourcepath}" srcdir="${test.src.dir}">**

**<customize>**

**<compilerarg line="${javac.test.compilerargs}"/>**

**</customize>**

**</j2seproject3:javac>**

**<copy todir="${build.test.classes.dir}">**

**<fileset dir="${test.src.dir}" excludes="${build.classes.excludes},${excludes}" includes="${includes}"/>**

**</copy>**

**</target>**

**<target name="-post-compile-test">**

**<!-- Empty placeholder for easier customization. -->**

**<!-- You can override this target in the ../build.xml file. -->**

**</target>**

**<target depends="init,compile,-pre-pre-compile-test,-pre-compile-test,-do-compile-test,-post-compile-test" name="compile-test"/>**

**<target name="-pre-compile-test-single">**

**<!-- Empty placeholder for easier customization. -->**

**<!-- You can override this target in the ../build.xml file. -->**

**</target>**

**<target depends="init,deps-jar,compile,-init-test-module-properties,-pre-pre-compile-test,-pre-compile-test-single" if="have.tests" name="-do-compile-test-single">**

**<fail unless="javac.includes">Must select some files in the IDE or set javac.includes</fail>**

**<j2seproject3:force-recompile destdir="${build.test.classes.dir}"/>**

**<j2seproject3:javac apgeneratedsrcdir="${build.test.classes.dir}" classpath="${javac.test.classpath}" debug="true" destdir="${build.test.classes.dir}" excludes="" includes="${javac.includes}, module-info.java" modulepath="${javac.test.modulepath}" processorpath="${javac.test.processorpath}" sourcepath="${test.src.dir}" srcdir="${test.src.dir}">**

**<customize>**

**<compilerarg line="${javac.test.compilerargs}"/>**

**</customize>**

**</j2seproject3:javac>**

**<copy todir="${build.test.classes.dir}">**

**<fileset dir="${test.src.dir}" excludes="${build.classes.excludes},${excludes}" includes="${includes}"/>**

**</copy>**

**</target>**

**<target name="-post-compile-test-single">**

**<!-- Empty placeholder for easier customization. -->**

**<!-- You can override this target in the ../build.xml file. -->**

**</target>**

**<target depends="init,compile,-pre-pre-compile-test,-pre-compile-test-single,-do-compile-test-single,-post-compile-test-single" name="compile-test-single"/>**

**<!--**

**=======================**

**TEST EXECUTION SECTION**

**=======================**

**-->**

**<target depends="init" if="have.tests" name="-pre-test-run">**

**<mkdir dir="${build.test.results.dir}"/>**

**</target>**

**<target depends="init,compile-test,-init-test-run-module-properties,-pre-test-run" if="have.tests" name="-do-test-run">**

**<j2seproject3:test includes="${includes}" testincludes="\*\*/\*Test.java"/>**

**</target>**

**<target depends="init,compile-test,-pre-test-run,-do-test-run" if="have.tests" name="-post-test-run">**

**<fail if="tests.failed" unless="ignore.failing.tests">Some tests failed; see details above.</fail>**

**</target>**

**<target depends="init" if="have.tests" name="test-report"/>**

**<target depends="init" if="netbeans.home+have.tests" name="-test-browse"/>**

**<target depends="init,compile-test,-pre-test-run,-do-test-run,test-report,-post-test-run,-test-browse" description="Run unit tests." name="test"/>**

**<target depends="init" if="have.tests" name="-pre-test-run-single">**

**<mkdir dir="${build.test.results.dir}"/>**

**</target>**

**<target depends="init,compile-test-single,-init-test-run-module-properties,-pre-test-run-single" if="have.tests" name="-do-test-run-single">**

**<fail unless="test.includes">Must select some files in the IDE or set test.includes</fail>**

**<j2seproject3:test excludes="" includes="${test.includes}" testincludes="${test.includes}"/>**

**</target>**

**<target depends="init,compile-test-single,-pre-test-run-single,-do-test-run-single" if="have.tests" name="-post-test-run-single">**

**<fail if="tests.failed" unless="ignore.failing.tests">Some tests failed; see details above.</fail>**

**</target>**

**<target depends="init,compile-test-single,-init-test-run-module-properties,-pre-test-run-single,-do-test-run-single,-post-test-run-single" description="Run single unit test." name="test-single"/>**

**<target depends="init,compile-test-single,-pre-test-run-single" if="have.tests" name="-do-test-run-single-method">**

**<fail unless="test.class">Must select some files in the IDE or set test.class</fail>**

**<fail unless="test.method">Must select some method in the IDE or set test.method</fail>**

**<j2seproject3:test excludes="" includes="${javac.includes}" testincludes="${test.class}" testmethods="${test.method}"/>**

**</target>**

**<target depends="init,compile-test-single,-pre-test-run-single,-do-test-run-single-method" if="have.tests" name="-post-test-run-single-method">**

**<fail if="tests.failed" unless="ignore.failing.tests">Some tests failed; see details above.</fail>**

**</target>**

**<target depends="init,compile-test-single,-init-test-run-module-properties,-pre-test-run-single,-do-test-run-single-method,-post-test-run-single-method" description="Run single unit test." name="test-single-method"/>**

**<!--**

**=======================**

**TEST DEBUGGING SECTION**

**=======================**

**-->**

**<target depends="init,compile-test-single,-pre-test-run-single" if="have.tests" name="-debug-start-debuggee-test">**

**<fail unless="test.class">Must select one file in the IDE or set test.class</fail>**

**<j2seproject3:test-debug excludes="" includes="${javac.includes}" testClass="${test.class}" testincludes="${javac.includes}"/>**

**</target>**

**<target depends="init,compile-test-single,-pre-test-run-single" if="have.tests" name="-debug-start-debuggee-test-method">**

**<fail unless="test.class">Must select one file in the IDE or set test.class</fail>**

**<fail unless="test.method">Must select some method in the IDE or set test.method</fail>**

**<j2seproject3:test-debug excludes="" includes="${javac.includes}" testClass="${test.class}" testMethod="${test.method}" testincludes="${test.class}" testmethods="${test.method}"/>**

**</target>**

**<target depends="init,compile-test" if="netbeans.home+have.tests" name="-debug-start-debugger-test">**

**<j2seproject1:nbjpdastart classpath="${debug.test.classpath}" name="${test.class}"/>**

**</target>**

**<target depends="init,compile-test-single,-init-test-run-module-properties,-debug-start-debugger-test,-debug-start-debuggee-test" name="debug-test"/>**

**<target depends="init,compile-test-single,-init-test-run-module-properties,-debug-start-debugger-test,-debug-start-debuggee-test-method" name="debug-test-method"/>**

**<target depends="init,-pre-debug-fix,compile-test-single" if="netbeans.home" name="-do-debug-fix-test">**

**<j2seproject1:nbjpdareload dir="${build.test.classes.dir}"/>**

**</target>**

**<target depends="init,-pre-debug-fix,-do-debug-fix-test" if="netbeans.home" name="debug-fix-test"/>**

**<!--**

**=========================**

**APPLET EXECUTION SECTION**

**=========================**

**-->**

**<target depends="init,compile-single" name="run-applet">**

**<fail unless="applet.url">Must select one file in the IDE or set applet.url</fail>**

**<j2seproject1:java classname="sun.applet.AppletViewer">**

**<customize>**

**<arg value="${applet.url}"/>**

**</customize>**

**</j2seproject1:java>**

**</target>**

**<!--**

**=========================**

**APPLET DEBUGGING SECTION**

**=========================**

**-->**

**<target depends="init,compile-single" if="netbeans.home" name="-debug-start-debuggee-applet">**

**<fail unless="applet.url">Must select one file in the IDE or set applet.url</fail>**

**<j2seproject3:debug classname="sun.applet.AppletViewer">**

**<customizeDebuggee>**

**<arg value="${applet.url}"/>**

**</customizeDebuggee>**

**</j2seproject3:debug>**

**</target>**

**<target depends="init,compile-single,-debug-start-debugger,-debug-start-debuggee-applet" if="netbeans.home" name="debug-applet"/>**

**<!--**

**===============**

**CLEANUP SECTION**

**===============**

**-->**

**<target name="-deps-clean-init" unless="built-clean.properties">**

**<property location="${build.dir}/built-clean.properties" name="built-clean.properties"/>**

**<delete file="${built-clean.properties}" quiet="true"/>**

**</target>**

**<target if="already.built.clean.${basedir}" name="-warn-already-built-clean">**

**<echo level="warn" message="Cycle detected: Banking Management System was already built"/>**

**</target>**

**<target depends="init,-deps-clean-init" name="deps-clean" unless="no.deps">**

**<mkdir dir="${build.dir}"/>**

**<touch file="${built-clean.properties}" verbose="false"/>**

**<property file="${built-clean.properties}" prefix="already.built.clean."/>**

**<antcall target="-warn-already-built-clean"/>**

**<propertyfile file="${built-clean.properties}">**

**<entry key="${basedir}" value=""/>**

**</propertyfile>**

**</target>**

**<target depends="init" name="-do-clean">**

**<delete dir="${build.dir}"/>**

**<delete dir="${dist.jlink.output}"/>**

**<delete dir="${dist.dir}" followsymlinks="false" includeemptydirs="true"/>**

**</target>**

**<target name="-post-clean">**

**<!-- Empty placeholder for easier customization. -->**

**<!-- You can override this target in the ../build.xml file. -->**

**</target>**

**<target depends="init,deps-clean,-do-clean,-post-clean" description="Clean build products." name="clean"/>**

**<target name="-check-call-dep">**

**<property file="${call.built.properties}" prefix="already.built."/>**

**<condition property="should.call.dep">**

**<and>**

**<not>**

**<isset property="already.built.${call.subproject}"/>**

**</not>**

**<available file="${call.script}"/>**

**</and>**

**</condition>**

**</target>**

**<target depends="-check-call-dep" if="should.call.dep" name="-maybe-call-dep">**

**<ant antfile="${call.script}" inheritall="false" target="${call.target}">**

**<propertyset>**

**<propertyref prefix="transfer."/>**

**<mapper from="transfer.\*" to="\*" type="glob"/>**

**</propertyset>**

**</ant>**

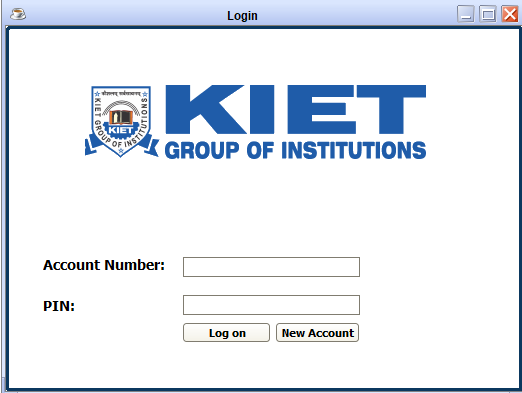
**</target>**

**</project>**

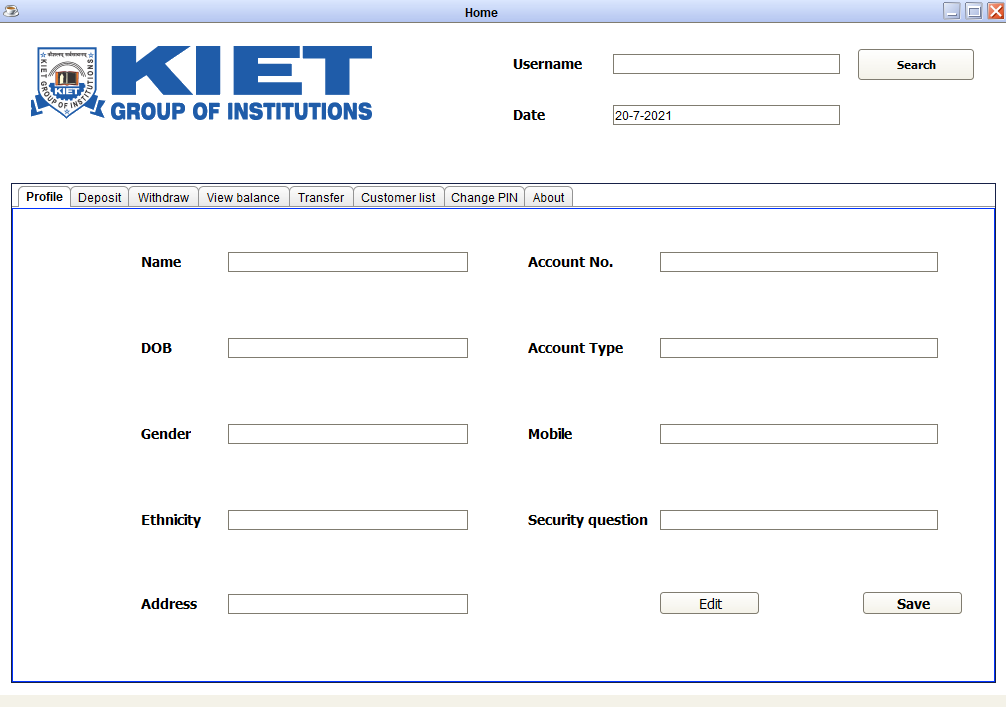
## CHAPTER 4

**OUTPUT SCREEN**

## Screenshots of login Page



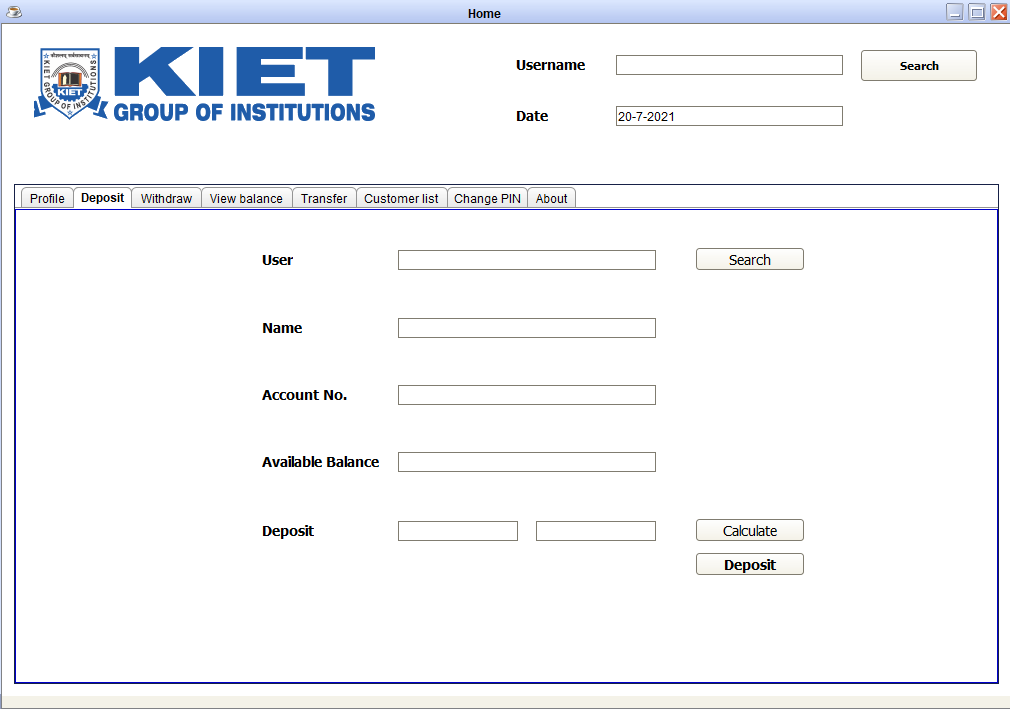
## Profile options

****

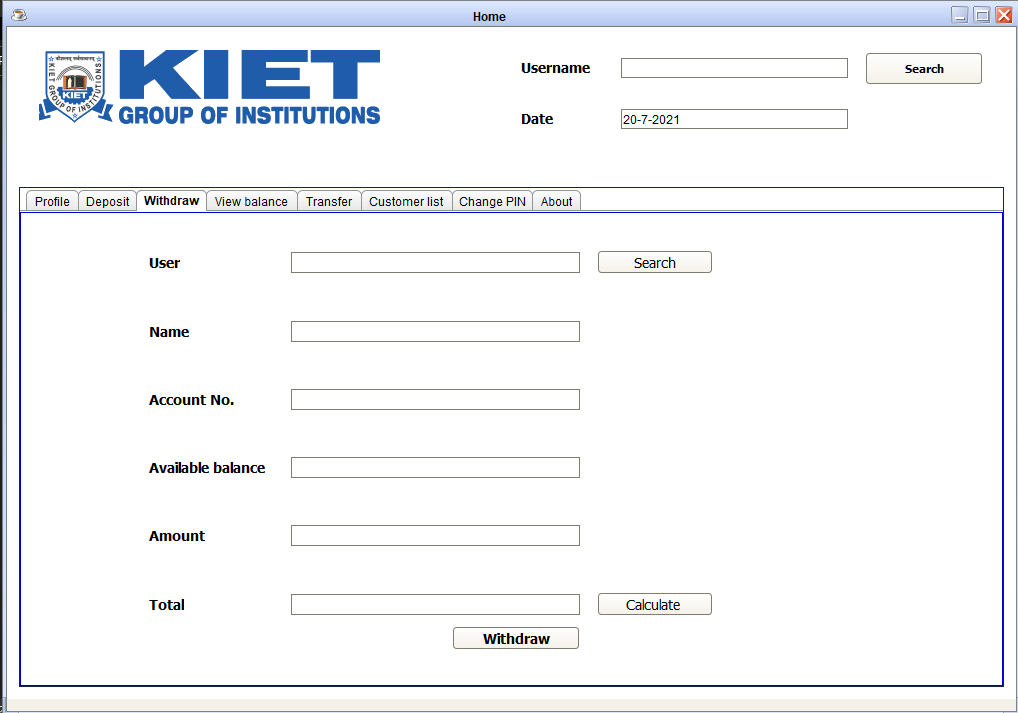
## 4.2.1 Editing existing profile

## 

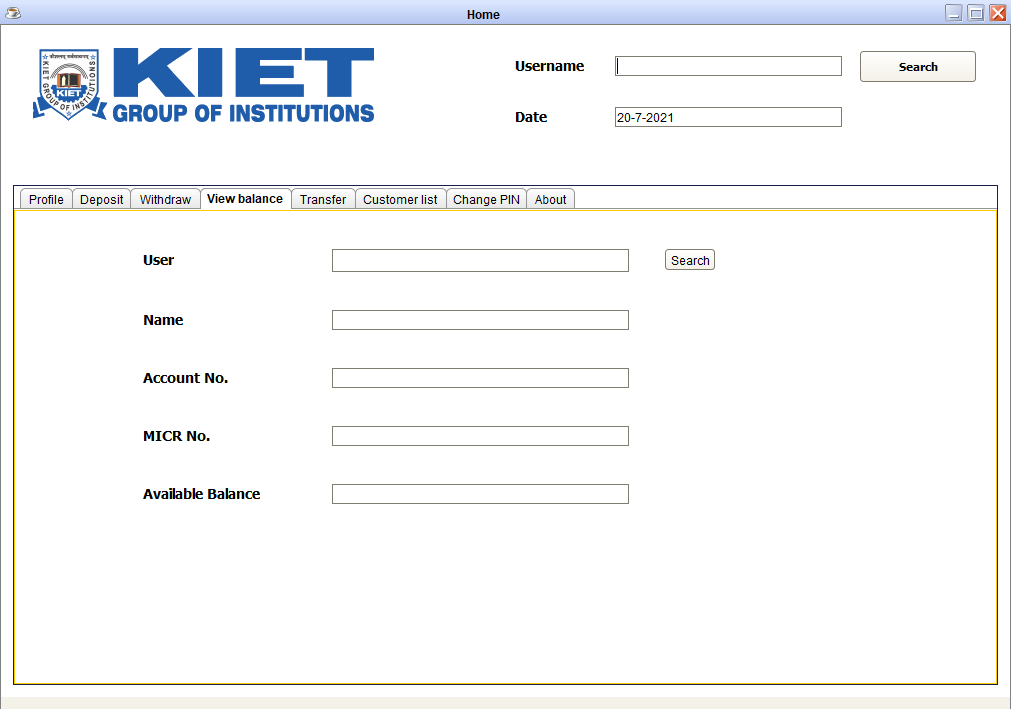
## Deposit form

****

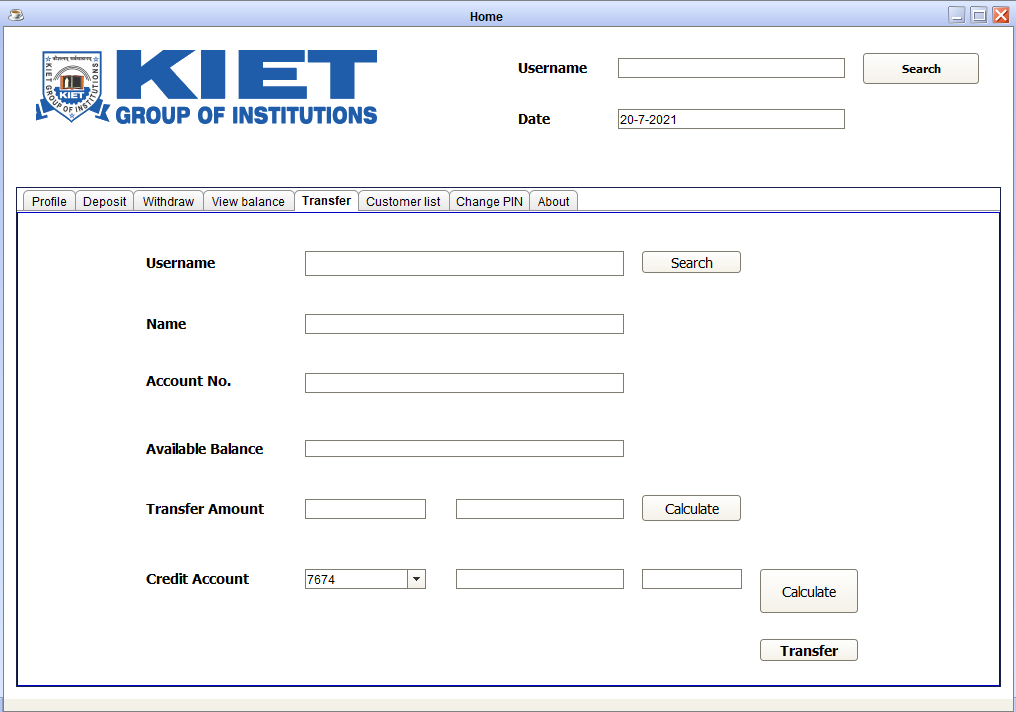
## 4.4 Withdraw form



## View Balance form

****

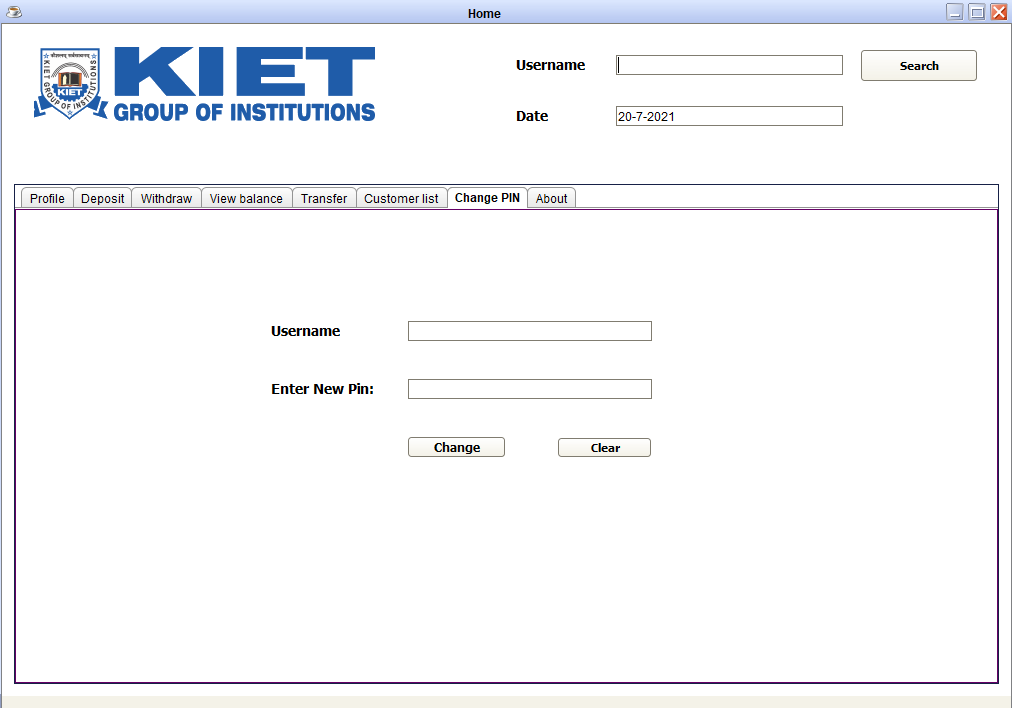
## Transfer form

****

## Customer list form

## 

## Change pin form

****

## About

## 

## References

1. Roy H. Campbell ; Charles A. Kamhoua ; Kevin A. Kwiat,2018
2. [Mahmud Hossain](https://ieeexplore.ieee.org/author/37086088790) ; [Rasib Khan](https://ieeexplore.ieee.org/author/37077770300) ; [Shahid Al Noor](https://ieeexplore.ieee.org/author/37085375878) ; [Ragib Hasan](https://ieeexplore.ieee.org/author/37419422100);2016
3. [Takayuki Kushida](https://ieeexplore.ieee.org/author/37339943800) ; [Gopal S. Pingali](https://ieeexplore.ieee.org/author/38179762100);2014
4. [Pierfrancesco Bellini](https://ieeexplore.ieee.org/author/37282956900) ; [Daniele Cenni](https://ieeexplore.ieee.org/author/37827445700) ; [Paolo Nesi](https://ieeexplore.ieee.org/author/37085359261);2015
5. [Tessema Mengistu](https://ieeexplore.ieee.org/author/37086157374) ; [Abdulrahman Alahmadi](https://ieeexplore.ieee.org/author/37085458866) ; [Abdullah Albuali](https://ieeexplore.ieee.org/author/37086151034) ; Yousef Alsenani ; [Dunren Che](https://ieeexplore.ieee.org/author/38247182800);2017
6. [Aobing Sun](https://ieeexplore.ieee.org/author/38235846500) ; [Guohong Gao](https://ieeexplore.ieee.org/author/37086523041) ; [Tongkai Ji](https://ieeexplore.ieee.org/author/37706792200) ; [Xuping Tu](https://ieeexplore.ieee.org/author/37086521240); 2018
7. [Fan Jing Meng](https://ieeexplore.ieee.org/author/37722776400) ; [Jian Wang](https://ieeexplore.ieee.org/author/37716395000) ; [Changhua Sun](https://ieeexplore.ieee.org/author/38195225900) ; [Dong Xu Duan](https://ieeexplore.ieee.org/author/38467639400) ; [Yi-Min Chee](https://ieeexplore.ieee.org/author/37328090900);2012
8. [Sodam Baek](https://ieeexplore.ieee.org/author/37085377987) ; [Kibae Kim](https://ieeexplore.ieee.org/author/37406159900) ; [Jörn Altmann](https://ieeexplore.ieee.org/author/37270584900), 2014
9. Anuradha Manchar;[Ankit Chouhan](https://ieeexplore.ieee.org/author/37085997823);2017
10. [Aneta Poniszewska-Maranda](https://ieeexplore.ieee.org/author/38272891100) ; [Radosław Matusiak](https://ieeexplore.ieee.org/author/37086047150) ; [Natalia Kryvinska](https://ieeexplore.ieee.org/author/37297240400);2017
11. [Jigar Patel](https://ieeexplore.ieee.org/author/37086003992) ; [Ankit Chouhan](https://ieeexplore.ieee.org/author/37085997823); **2016**
12. L.N. Goslin ; C.L. Petersen;1997
13. [Kristina Ozimec](https://ieeexplore.ieee.org/author/37085956110) ; [Marija Tomaševic Lišanin](https://ieeexplore.ieee.org/author/37085956783);2011
14. [Jun Wang](https://ieeexplore.ieee.org/author/37406469100) ; [Moninder Singh](https://ieeexplore.ieee.org/author/37067542000) ; [Kush R. Varshney](https://ieeexplore.ieee.org/author/37709490300);2012
15. [Kush R. Varshney](https://ieeexplore.ieee.org/author/37709490300) ; [Moninder Singh](https://ieeexplore.ieee.org/author/37067542000);2013.