Q1.

A java applet is a special kind of java application that is designed to work within a web browser. For safety and security they are not permitted to perform file or certain memory operations on the client machine, this is to prevent applet from delete files on the clients machine. Java applets are usually compact, so that they can easily downloaded from the web.

Applets can do many things which are not possible by html alone. Applets can talk to the browser via javascript. In our case we have been using applets to simulate and program Client Server interactions where the server sends an interface(e.g. calculator interface) and when the user inputs numbers, that data is sent back to the server program and it performs the calculations on the server end before sending the result back to the user. Therefore the client doesn’t use many resources by doing the calculation themselves.

This is a very basic example of how applets are used. In general there is no limit to what applets are limited by.

Q2.

package Bookstore;  
  
import java.rmi.registry.\*;  
public class BookClient{  
  
 public static void main(String[] args) {  
 try {  
 Registry reg = LocateRegistry.*getRegistry*("localhost");  
 Book stub = (Book) reg.lookup("Book");  
 System.*out*.println(stub.price(Bookname));  
 System.*out*.println(stub.payment(amount));  
 System.*out*.println(stub.searchBook(BookName));  
 }  
 catch(Exception e ) {System.*out*.println(e);}  
  
 }  
  
}

package Bookstore;  
  
import java.rmi.RemoteException;  
import java.rmi.server.UnicastRemoteObject;  
import java.rmi.registry.\*;  
public class BookServer implements Book{  
 public String Price(String BookName) {  
 return Bookname.value;  
 }  
  
 @Override  
 public String price(String BookName) throws RemoteException {  
 return null;  
 }  
  
 public String payment(double amount) {  
 if(payment()) {  
 return "payment successful";  
 }  
 return "payment failed";  
 }  
  
 private boolean payment() {  
 return false;  
 }  
  
 public String SearchBook(String BookName) {  
 if(Searchbook()) {  
 return BookName+" is found";  
 }  
 return BookName+" not found";  
 }  
  
 private boolean Searchbook() {  
  
 }  
  
 @Override  
 public String searchBook(String BookName) throws RemoteException {  
 // *TODO Auto-generated method stub* return null;  
 }  
 public static void main(String[] args) {  
 try {  
 BookServer obj = new BookServer();  
 Book stub = (Book) UnicastRemoteObject.*exportObject*(obj,0);  
 Registry reg = LocateRegistry.*getRegistry*();  
 reg.bind("Book",stub);  
 System.*out*.println("Server is running");  
 }  
 catch(Exception e) {System.*out*.println(e);}  
 }  
}

package Bookstore;  
  
import java.rmi.\*;  
public interface Book extends Remote{  
 String price(String BookName) throws RemoteException;  
 String payment(double amount) throws RemoteException;  
 String searchBook(String BookName) throws RemoteException;  
}

Q3.

import java.applet.\*;  
import javax.swing.\*;  
  
public class Echo extends Applet {  
 TextArea output;  
 TextField input;  
 String s;  
 Checkbox escape;  
 boolean bool=true;  
 String args;  
   
 public void init () {  
 setLayout(new BorderLayout());  
 final Applet Echo = this;  
 s = "enter some text";  
 input = new TextField(s);  
 this.add(input,BorderLayout.SOUTH);  
 output = new TextArea("");  
 this.add(output,BorderLayout.CENTER);  
 }  
   
 public class Listener implements ActionListener, ItemListener {  
 public void actionPerformed(ActionEvent e) {  
 args = this.getParameter("parameter");  
 input.getText();  
 output.setText(input.getText());  
 if (bool) {  
 System.*out*.println(args.replaceAll("\\\\n", "\n").replaceAll("\\\\t","\t"));  
 } else {  
 System.*out*.println(args);  
 }  
 }  
  
 private String getParameter(String parameter) {  
 return parameter;  
 }  
  
 public void itemStateChanged(ItemEvent ie) {  
 if(!escape.getState())  
 bool=false;  
 }  
 }  
}

Q4.

import java.awt.Graphics;  
import javax.swing.\*;  
public class Calculator extends JApplet{  
 double sum;  
 public void init(){  
 String firstNumber;  
 String secondNumber;  
 double number1;  
 double number2;  
 firstNumber = JOptionPane.*showInputDialog*("Enter first Number");  
 secondNumber = JOptionPane.*showInputDialog*("Enter second Number");  
 number1 = Double.*parseDouble*(firstNumber);  
 number2 = Double.*parseDouble*(secondNumber);  
 sum = number1 + number2;  
 }  
  
 public void paint (Graphics g){  
 super.paint( g );  
 g.drawRect(15, 10, 270, 20);  
 g.drawString("The sum is " + sum, 25, 25);  
 }  
}