```java
/**
This is the server which is responsible for binding remote objects
into the rmi Registry and starts it.
*/
import java.rmi.*;
import java.rmi.registry.*;

public class Server{
 private static String HOST_NAME = "localhost";
 private static Registry registry;
 private static String registry_port = "1099";
 private static String remote_object;

 //constructor that binds the remote_object to the rmiRegistry with default name
"Adder"
 public Server()throws Exception{
    this("Adder");
    Addimp remote_obj = new Addimp();

Naming.rebind("rmi://"+HOST_NAME+":"+registry_port+"/"+remote_object,remote_obj);
    System.out.println("Server is running.");
    }

 //constructor with a parameter as name of the remote object
 public Server(String remote_object){
  this.remote_object = remote_object;
 }

 //registry must be started before binding an object into it.
 public  static void startRegistry(int port)throws Exception{
    registry = LocateRegistry.createRegistry(Integer.parseInt(registry_port));
 }

 public static void main(String...args)throws Exception{
        /*Remember:
          RMI's class loader will download classes from remote locations
          only if a security manager has been set.
          if(System.getSecurityManager==null)
            System.setSecurityManager(new java.lang.securityManager());*/

          startRegistry(Integer.parseInt(registry_port));

          //server is set up
          new Server();
 }
}
```