

```

import java.io.PrintWriter;

import java.sql.ResultSet;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

////////////////////////////////////
//                                     //
//                                     ALQUILER                                     //
//                                     //
////////////////////////////////////

/**
 * Servlet implementation class DimeOcupacionPuestosCercanos
 */
public class OcupacionPuestosCercanos extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public OcupacionPuestosCercanos() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse re
sponse)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse respons
e) throws ServletException, IOException {
        // TODO Auto-generated method stub
        doPost(request, response);
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse r
esponse)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse respon
se) throws ServletException, IOException {

        String pagOK="";

        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        HttpSession session = request.getSession(true);

        conexionBD consulta = new conexionBD();
        String numPuesto="";
        Integer maxBicis=0;

```

```

String idPuesto = request.getParameter("puesto");
String idAccion = request.getParameter("accion");
if (idAccion==null ) idAccion = "disponibles";
if (! ( idAccion.equals("disponibles") || idAccion.equals("libres") ) ){

    out.println(noExistenCercanos(""));
    out.close();
    consulta.cerrarConexion();
    return;
}

//comprobamos si el parametro es null o no existe el puesto
if (! consulta.existePuesto(idPuesto)){
    out.println(noExistenCercanos(""));
    out.close();
    consulta.cerrarConexion();
    return;
}
if (idAccion.equals("disponibles"))
    pagOK += Cadenas.htmlIni("Puestos Disponibles",idPuesto, "", "", "");
else
    pagOK += Cadenas.htmlIni("Puestos Libres",idPuesto, "", "", "");
pagOK += Cadenas.htmlIniDivCentral();

////////////////////////////////////
// Conexi3n y obtenci3n de resultados con la base de datos //
////////////////////////////////////
try
{

    GestionLogs.escribirLog(sesion, "Buscando Puestos cercanos al puesto
("+idPuesto+)");
    String listaPuestosCercanos=consulta.dimePuestosCercanos(idPuesto);
//+*+//Cadenas.alert("lista puestos cercanos a ("+idPuesto+"): "+listaPuestosCer
canos);

    String puestosCercanos[] = listaPuestosCercanos.split(",");
    GestionLogs.escribirLog(sesion, "Puestos cercanos al puesto ("+idPue
sto+)"="+listaPuestosCercanos);
    if(puestosCercanos.length==0){
        out.println(noExistenCercanos(idPuesto));
        out.close();
        consulta.cerrarConexion();
        return;
    }
    else{
        pagOK += "<center><font size=+2><b>Puesto "+idPuesto+"</b></font
> \n";

        if (idAccion.equals("disponibles")){
            //pagOK += "<br> No hay disponibles";
            pagOK += "<br>SIN Bicicletas<br><i><center>bicicletas en <br
>puestos cercanos</i><br></center> \n";
        }else{
            //pagOK += "<br> imposible";
            pagOK += "<br>SIN Sitios<br><i><center>sitios libres en <br>
puestos cercanos</i><br></center> \n";
        }
        ///pagOK += "<div class='miDivTabla'> \n";
        pagOK += "    <center> <table margin-top='50px' width=200px border
=1 cellpadding='0' cellspacing='5' align='center'>\n";

```

```

        if (idAccion.equals("disponibles"))
            pagOK += " <tr><th class='miTH' width='60px'>puesto</th><th
class='miTH'><b>disponibles</b> </th></tr>\n";
        else
            pagOK += " <tr><th class='miTH' width='60px'>puesto</th><th
class='miTH'><b>libres</b> </th></tr>\n";

        for (int i=0;i<puestosCercanos.length;i++){
            numPuesto = puestosCercanos[i];
            /*AQUIAQUI*/ //if (true){
            if (consulta.existePuesto(numPuesto)){
                /*AQUIAQUI*/ //if (true){
                if ( consulta.existePuestoEnBiciPuestos(numPuesto)){
                    int maxBicisPuesto=consulta.dimeNumMaxBicisEnPuesto(
numPuesto);

                    int numBicisEnPuesto=consulta.numeroBicisOcupadasEnP
uesto(numPuesto);

                    if (maxBicisPuesto>0){
                        if (idAccion.equals("disponibles"))
                            pagOK += crearFila(numPuesto,numBicisEnPuest
o);

                        else
                            pagOK += crearFila(numPuesto,maxBicisPuesto-
numBicisEnPuesto);
                    }
                    else{
                        pagOK += crearFilaError(numPuesto,"inf. no dispo
nible 1");

                        GestionLogs.escribirLog(sesion, "ERROR : Puesto(
"+numPuesto+") mx_bicis="+maxBicisPuesto);
                    }
                }else{
                    //++// Cadenas.alert("puesto("+numPuesto+") NO EXIS
TE");

                    pagOK += crearFilaError(numPuesto,"inf. no disponibl
e 2");

                    GestionLogs.escribirLog(sesion, "ERROR : NO EXISTE
PUESTO en Bicis_Puesto("+numPuesto+")");
                }
            }
            else{
                pagOK += crearFilaError(numPuesto, " inf. no disponible
3");

                GestionLogs.escribirLog(sesion, "ERROR : Puesto("+numPue
sto+") No Encontrado");
            }
        }
    }
}
catch (Exception e)
{
    GestionLogs.escribirLog(sesion, "EXCEPCIÃN: " + e.toString());
    pagOK += "<h3 align='center'>OcurriÃ³ un error interno</h3>";
    pagOK += "</BODY></HTML>";
    out.println(pagOK);
    out.close();
    consulta.cerrarConexion();
    return;
}
}

```

```

        pagOK += "</table></center>\n";
        ///pagOK += "</div>\n"; //de 'miDivTabla'> \n";
        pagOK += Cadenas.htmlFinDivCentral();
        pagOK += Cadenas.htmlFin();
        out.println(pagOK);
        out.close();
        consulta.cerrarConexion();
        return;
    }

    private String crearFila(String numPuesto,int ocupadas)
    {
        String strFila = "";
        strFila = "<tr><td class='celdaPuesto'>" + numPuesto + "</td>";
        if (ocupadas==0)
            strFila += "<td class='celdaSinBicisLibres'>" + String.valueOf(ocupadas
) + "</td>";
        else
            strFila += "<td class='celdaConBicisLibres'>" + String.valueOf(ocupadas
) + "</td>";
        strFila += "</tr>\n";

        return strFila;
    }

    private String crearFilaError(String numPuesto,String txtError)
    {
        String strFila = "<tr><td class='celdaSinBicisLibres'>" + numPuesto + "</td
>" +
            "<td class='celdaError'>(" + txtError + ")</td>" +
            "</tr>\n";

        return strFila;
    }

    private String noExistenCercanos(String numPuesto){
        String pagR="";
        pagR =Cadenas.htmlIni("Puestos Libres","SIN PUESTO", "", "", "");
        pagR += Cadenas.htmlIniDivCentral();
        pagR += "<center><table width=200px border=1 cellspacing='0' cellpadding
='5' align='center'>\n";
        pagR += crearFilaError(numPuesto, "No Existen puestos cercanos");
        pagR += "</table>";
        pagR += Cadenas.htmlFinDivCentral();
        pagR += Cadenas.htmlFin();
        return pagR;
    }
}

```