om.uDistrital.avanzada.parcialUno.modelo.conexic **Animales Exoticos** ConexionBaseDatos -String URLBD m.uDistrital.avanzada.parcialUno.modelo.Interfaces -String usuario -String contrasena -String cn **ICreate** ______ + insertar(generico) + Connection getConexion():return **IRead** - desconectar(): + consultar(dato) **IModifier** ______ + modificar(mascota) om.uDistrital.avanzada.parcialUno.contro +eliminar(apodo) ControlRAF -RAFDAO rafDAO new ControlGeneral() boolean tieneDisponible() List leerTodas() ControlGeneral + void guardarTodas() -ControlProperties cProperties + void limpiar() m.uDistrital.avanzada.parcialUno.modelo.DAO m.uDistrital.avanzada.parcialUno.modelo -ControlRAF cRAF -ControlSerializacion cSerializacion -ControlAnimal cAnimal +void iniciarPrograma(): Inicializa el sistema y crea el ControlVista #String nombreComun +void aceptarMascotasCompletadas(completadas): Recibe y guarda las mascotas ArchivoSerializacion #String clasificacion #String familia +void registrarMascota(apodo, nombreComun, clasificacion, familia, genero, especie, -File archivo #String genero alimento): Registra una nueva mascota MascotaDAO propertiesDAO #String especie +void modificarMascota(apodo, nombreComun, clasificacion, familia, genero, especie, + ArchivoSerializacion(rura) #String alimentoPrincipal alimento): Modifica una mascota existente +ObjectOutputStream abrirEscritura():archivo -ArchivoProperties archivo -String TBL **ArchivoProperties** +void eliminarMascota(apodo): Elimina una mascota por apodo + File getArchivo() -String colApodo +MascotaVO consultarMascota(apodo): Retorna la mascota consultada -String colNombreComun MascotaVO consultar (apodo) -File archivo +List<MascotaVO> listarTodasLasMascotas(): Lista todas las mascotas registradas -String colClasificacion +list listarTodas () +Animal(nombreComun, clasificacion, familia, genero, `-----+List<MascotaVO> consultarPorClasificacion(clasificacion): Consulta por clasificación -String colFamilia -MascotaVo construitMascota ControlAnimal especie, alimentoPrincipal) +ArchivoProperties(ruta): +List<MascotaVO> consultarPorFamilia(familia): Consulta por familia -String colGenero +String safe (valor) +String getNombreComun(): nombreComun +InputStream open(): archivo abierto +List<MascotaVO> consultarPorTipoAlimentoVO(alimento): Consulta por tipo de alimento -String colEspecie +void setNombreComun(nombreComun) +File getArchivo(): archivo +void serializarMascotas(carpetaDestino): Serializa todas las mascotas en la ruta indicada -String colAlimento +String getClasificacion(): clasificacion +void guardarEstadoEnRAF(): Guarda el estado actual del sistema en el archivo RAF -----, +void setClasificacion(clasificacion) +void registar(apodo, nombreComun, clasificacion, familia, genero, especie, -void flujoCargalnicial(): Carga inicial del sistema (RAF o properties) +void insertar (mascota) +String getFamilia(): familia alimentoPrincipal) -void cargarMascotasABD(mascotas): Inserta en BD evitando duplicados **ArchivoRandomAccessFile** + MascotaVO consultar (apodo) +void setFamilia(familia) +Animal consultar(apodo): Mascota consultada + void consultar (mascota) +String getGenero(): genero +List<MascotaVo> listarTodas(): Lista de mascotas -File archivo + void eliminar (apodo) +void setGenero(genero) +void modificar(apodo, nombreComun, clasificacion, familia, genero, +list listarTodas () +String getEspecie(): especie especie, alimentoPrincipal) +list consultarPorClasificacion(clasificacion) +void setEspecie(especie) +ArchivoRandomAccessFile(ruta) +void eliminar(apodo): +list consultarPorFamilia(familia) +String getAlimentoPrincipal(): alimentoPrincipal +Lista<MascotaVO> consultarPorClasificacion(clasificacion): +RandomAccessFile abrir(modo): Archivo +list consultarPorTipoAlimento(tipoAlimeto) +void setAlimentoPrincipal(alimentoPrincipal) ConsultaClasificacion +boolean tieneDatos(): true or false +MascotaVO map (rs) ,----<u>-</u>---' +Lista<MascotaVO> consultarPorAlimentoPrincipal(clasificacion): +long tamanio():Longitud del archivo +File getArchivo() ConsultaAlimento ControlVista SerializacionDAO -void validarApodo(apodo) + void guardarMascotasEnBD(listaPropoBD) -String CMD_ADICIONAR -ArchivoRandomAccesFile archivoRAF -ArchivoSerializacion +List<MascotaVO> combinarListas(originales): listas unidas -String CMD_CONSULTAR -String rutaArchivo -String nombreArchivo +void procesarMascota(mascotas) -String CMD_MODIFICAR MascotaVO -String CMD_ELIMINAR +void insertar (mascota) +void insertar (mascotas) -String CMD_SERIALIZAR +guardarTodas (mascotas) +void serializarEnRuta(mascotas) -String CMD_LIMPIAR +coid serializarMascotas(mascotas) +list consultar(dato) -String CMD_SALIR -MascotaVO(apodo, nombreComun, clasificacion, familia, +list leerTodas() +String getNombreArchivo genero, especie, alimentoprincipal) +boleean tieneDatos() +ArchivoSerializacion getArchivoSerializacion() +ControlVista(cGeneral) +void limpiar() private void registrarListenersYComandos(): Vincula botones y +String getApodo(): apodo +long obtenerTamanio() void setApodo(apodo) -private boolean esVacio(s: String) : Verifica cadena vacía -private boolean validarObligatorios(): Valida campos requeridos -private void pintarLista(mascotas: List<MascotaVO>) : Muestra listado formateado -private void cargarDatosEnVista(m: MascotaVO) : Carga datos consultados en la vista ControlSerializacion -private String camposFaltantes(m: MascotaVO) : Identifica -SerializacionDao serDao campos incompletos -private void completarDatosInteractivo(incompletas: List<MascotaVO>): Pide datos faltantes a la vista +ControlSerializacion() +void mostrarMascotasIncompletas(incompletas: +void exportar(lista, carpetaDestino List<MascotaVO>) : Flujo de completado de datos -private void onAdicionar() : Acción botón "Adicionar" -private void onConsultar() : Acción botón "Consultar" -private void onModificar() : Acción botón "Modificar" ControlProperties -private void onEliminar() : Acción botón "Eliminar" -private void onSerializar() : Acción botón "Serializar" PropertiesDAO proDAO -private void onLimpiar() : Acción botón "Limpiar" -private void onSalir() : Acción botón "Salir" +List<MascotaVO> cargarTodas(): Carga todas las +void mostrarMensaje(msg: String) : Muestra mensaje informativo mascotas definidas en el archivo .properties (vía vista) +MascotaVO consultarPorApodo(apodo): Consulta una +void mostrarError(msg: String) : Muestra mensaje de error (vía mascota por su apodo en el archivo .properties +List<MascotaVO> filtrarIncompletas(lista): Retorna las mascotas con datos faltantes +boolean esIncompleta(mascota): Verifica si una mascota tiene campos vacíos ------–boolean vacio(cadena): Evalúa si una cadena está vacía o om.uDistrital.avanzada.parcialUno.vist VistaPrincipal -JTextField txtApodo : Campo de texto para ingresar el apodo de la mascota -JTextField txtNombreComun : Campo de texto para ingresar el nombre común -JTextField txtClasificacion : Campo de texto para ingresar la clasificación biológica -JTextField txtFamilia : Campo de texto para ingresar la familia -JTextField txtGenero : Campo de texto para ingresar el género -JTextField txtEspecie : Campo de texto para ingresar la especie -JTextField txtAlimento : Campo de texto para ingresar el alimento principal -JComboBox<String> cmbCriterioConsulta : Desplegable para seleccionar el criterio de búsqueda -JTextField txtValorConsulta : Campo de texto para ingresar el valor de búsqueda -JTextArea areaResultado : Área de texto donde se muestran los resultados de consultas o listados -JButton btnAdicionar : Botón para registrar una nueva mascota -JButton btnConsultar : Botón para consultar una mascota existente -JButton btnModificar : Botón para modificar los datos de una mascota -JButton btnEliminar : Botón para eliminar una mascota del sistema -JButton btnSerializar : Botón para exportar las mascotas mediante serialización -JButton btnLimpiar : Botón para limpiar todos los campos del formulario JButton btnSalir : Botón para cerrar la aplicación y guardar el estado en RAF -JPanel panelPrincipal : Panel contenedor de todos los componentes visuales JScrollPane scrollResultado : Panel de desplazamiento que contiene el área de resultados JLabel lblApodo : Etiqueta descriptiva del campo Apodo JLabel lblNombreComun : Etiqueta descriptiva del campo Nombre Común -JLabel IblClasificacion : Etiqueta descriptiva del campo Clasificación -JLabel IblFamilia : Etiqueta descriptiva del campo Familia -JLabel IblEspecie : Etiqueta descriptiva del campo Especie -JLabel lblAlimento : Etiqueta descriptiva del campo Alimento Principal -JLabel IblCriterio : Etiqueta descriptiva del campo de criterio de consulta -JLabel lblValor : Etiqueta descriptiva del campo de valor de consulta +void mostrarMensaie(mensaie, titulo, tipo); Muestra un mensaie genérico en pantalla +void mostrarResultado(texto): Despliega resultados o listados en el área de texto principal +String leerDato(prompt): Solicita un dato al usuario mediante un cuadro de entrada +void limpiarCampos(): Limpia todos los campos de texto del formulario +String getApodo(): Retorna el apodo ingresado en la interfaz +String getNombreComun(): Retorna el nombre común ingresado +String getClasificacion(): Retorna la clasificación seleccionada +String getFamilia(): Retorna la familia ingresada +String getGenero(): Retorna el género ingresado +String getEspecie(): Retorna la especie ingresada +String getAlimento(): Retorna el alimento principal ingresado +String getCriterioConsulta(): Retorna el criterio de búsqueda seleccionado +String getValorConsulta(): Retorna el valor escrito para la consulta +void setApodo(apodo): Establece el apodo en el campo correspondiente +void setNombreComun(nombre): Establece el nombre común en el campo correspondiente +void setClasificacion(clasificacion): Establece la clasificación en el campo correspondiente +void setFamilia(familia): Establece la familia en el campo correspondiente +void setGenero(genero): Establece el género en el campo correspondiente +void setEspecie(especie): Establece la especie en el campo correspondiente +void asignarListenerAdicionar(listener): Asocia un ActionListener al botón Adicionar +void asignarListenerConsultar(listener): Asocia un ActionListener al botón Consultar +void asignarListenerModificar(listener): Asocia un ActionListener al botón Modificar +void asignarListenerEliminar(listener): Asocia un ActionListener al botón Eliminar +void asignarListenerSerializar(listener): Asocia un ActionListener al botón Serializar +void asignarListenerLimpiar(listener): Asocia un ActionListener al botón Limpiar +void asignarListenerSalir(listener): Asocia un ActionListener al botón Salir +void asignarActionCommandAdicionar(cmd): Define el ActionCommand del botón Adicionar +void asignarActionCommandConsultar(cmd): Define el ActionCommand del botón Consultar +void asignarActionCommandModificar(cmd): Define el ActionCommand del botón Modificar +void asignarActionCommandEliminar(cmd): Define el ActionCommand del botón Eliminar +void asignarActionCommandSerializar(cmd): Define el ActionCommand del botón Serializar +void asignarActionCommandLimpiar(cmd): Define el ActionCommand del botón Limpiar +void asignarActionCommandSalir(cmd): Define el ActionCommand del botón Salir +void info(mensaje): Muestra un mensaje informativo en un JOptionPane +void error(mensaje): Muestra un mensaje de error en un JOptionPane +void warn(mensaje): Muestra una advertencia en un JOptionPane +boolean confirmar(mensaje): Solicita confirmación del usuario y retorna su respuesta +String pedirDato(prompt): Pide al usuario ingresar un valor de texto -String elegirCarpetaSerializacion(): Abre un JFileChooser para seleccionar carpeta de destino

`