**package** edu.cnm.deepdive.healthtracker;  
  
  
**import** android.os.Bundle;  
**import** android.support.design.widget.NavigationView.OnNavigationItemSelectedListener;  
**import** android.support.v4.app.DialogFragment;  
**import** android.support.v4.app.Fragment;  
**import** android.support.v4.app.FragmentManager;  
**import** android.support.v4.app.FragmentTransaction;  
**import** android.view.View;  
**import** android.support.design.widget.NavigationView;  
**import** android.support.v4.view.GravityCompat;  
**import** android.support.v4.widget.DrawerLayout;  
**import** android.support.v7.app.ActionBarDrawerToggle;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.support.v7.widget.Toolbar;  
**import** android.view.Menu;  
**import** android.view.MenuItem;  
**import** android.widget.AdapterView;  
**import** android.widget.AdapterView.OnItemSelectedListener;  
**import** android.widget.ArrayAdapter;  
**import** android.widget.Spinner;  
**import** com.j256.ormlite.android.apptools.OpenHelperManager;  
**import** com.j256.ormlite.dao.Dao;  
**import** com.j256.ormlite.stmt.QueryBuilder;  
**import** edu.cnm.deepdive.healthtracker.entities.Patient;  
**import** edu.cnm.deepdive.healthtracker.helpers.OrmHelper;  
**import** edu.cnm.deepdive.healthtracker.loginfragments.CreatePatientFragment;  
**import** edu.cnm.deepdive.healthtracker.mainfragments.ListAllergyFragment;  
**import** edu.cnm.deepdive.healthtracker.mainfragments.ListHospitalizationFragment;  
**import** edu.cnm.deepdive.healthtracker.mainfragments.ListImmunizationFragment;  
**import** edu.cnm.deepdive.healthtracker.mainfragments.ListMedicationFragment;  
**import** edu.cnm.deepdive.healthtracker.mainfragments.MedicationFragment;  
**import** edu.cnm.deepdive.healthtracker.mainfragments.ListOfficeVisitFragment;  
**import** java.sql.SQLException;  
**import** java.util.ArrayList;  
**import** java.util.List;  
  
  
**public class** MainActivity **extends** AppCompatActivity  
 **implements** OnNavigationItemSelectedListener, OrmHelper.OrmInteraction {  
  
 **public static final** String ***PATIENT\_ID\_KEY*** = **"patient\_id"**;  
  
 **private** OrmHelper **helper** = **null**;  
 **private int patientSelected**;  
 **private** Patient **selectedPatient** = **null**;  
 **private** Spinner **spinner**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
  
 getHelper().getWritableDatabase().close();  
 setContentView(R.layout.***activity\_main***);  
  
 Toolbar toolbar = (Toolbar) findViewById(R.id.***toolbar***);  
 setSupportActionBar(toolbar);  
  
 *//get the spinner from the xml* **spinner**= (Spinner) findViewById(R.id.***name\_spinner***);  
 **spinner**.setOnItemSelectedListener(**new** OnItemSelectedListener() {  
 @Override  
 **public void** onItemSelected(AdapterView<?> adapterView, View view, **int** i, **long** l) {  
 **selectedPatient** = (Patient) adapterView.getItemAtPosition(i);  
 **if** (**selectedPatient**.getId() == 0) {  
 DialogFragment dialogFragment = **new** CreatePatientFragment();  
 dialogFragment.show(getSupportFragmentManager(), **"createPatient"**);  
 }  
  
 }  
  
  
 @Override  
 **public void** onNothingSelected(AdapterView<?> adapterView) {  
  
 }  
 });  
  
  
  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 ActionBarDrawerToggle toggle = **new** ActionBarDrawerToggle(  
 **this**, drawer, toolbar, R.string.***navigation\_drawer\_open***, R.string.***navigation\_drawer\_close***);  
 drawer.setDrawerListener(toggle);  
 drawer.openDrawer(GravityCompat.***START***);*//Set the drawer to open at start* toggle.syncState();  
  
 NavigationView navigationView = (NavigationView) findViewById(R.id.***nav\_view***);  
 navigationView.setNavigationItemSelectedListener(**this**);  
  
 addItemsOnSpinner();  
 }  
  
 **public void** addItemsOnSpinner(){  
 **try** {  
 Dao<Patient, Integer> dao = getHelper().getPatientDao();  
 QueryBuilder<Patient, Integer> builder = dao.queryBuilder();  
 builder.orderBy(**"NAME"**, **true**);  
 List<Patient> patients = dao.query(builder.prepare());  
 Patient addPatient = **new** Patient();  
 addPatient.setName(**"Add New Patient"**);  
 patients.add(addPatient);  
 *//create an adapter to describe how the items are displayed* ArrayAdapter<Patient> adapter = **new** ArrayAdapter<>(**this**,  
 android.R.layout.***simple\_spinner\_dropdown\_item***, patients);  
 *//set the spinners adapter to the previously created one.* **spinner**.setAdapter(adapter);  
 **if** (patients.size() > 0) {  
 **selectedPatient** = patients.get(0);  
 }  
 } **catch** (SQLException e) {  
 **throw new** RuntimeException(e);  
 }  
 }  
  
 @Override  
 **protected void** onStart() {  
 **super**.onStart();  
 getHelper();  
 }  
  
 @Override  
 **protected void** onStop() {  
 releaseHelper();  
 **super**.onStop();  
 }  
  
 *//creates an instance of OrmHelper* @Override  
 **public synchronized** OrmHelper getHelper() {  
 **if** (**helper** == **null**) {  
 **helper** = OpenHelperManager.*getHelper*(**this**, OrmHelper.**class**);  
 }  
 **return helper**;  
 }  
  
 *// prevents memory leaks by setting the helper to null when not in use* **public synchronized void** releaseHelper() {  
 **if** (**helper** != **null**) {  
 OpenHelperManager.*releaseHelper*();  
 **helper** = **null**;  
 }  
 }  
  
 @Override  
 **public void** onBackPressed() {  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 **if** (drawer.isDrawerOpen(GravityCompat.***START***)) {  
 drawer.closeDrawer(GravityCompat.***START***);  
 } **else** {  
 **super**.onBackPressed();  
 }  
 }  
  
 @Override  
 **public boolean** onCreateOptionsMenu(Menu menu) {  
 *// Inflate the menu; this adds items to the action bar if it is present.* getMenuInflater().inflate(R.menu.***main***, menu);  
 **return true**;  
 }  
  
 @Override  
 **public boolean** onOptionsItemSelected(MenuItem item) {  
 *// Handle action bar item clicks here. The action bar will  
 // automatically handle clicks on the Home/Up button, so long  
 // as you specify a parent activity in AndroidManifest.xml.* **int** id = item.getItemId();  
  
 *//noinspection SimplifiableIfStatement* **if** (id == R.id.***action\_settings***) {  
 **return true**;  
 }  
  
 **return super**.onOptionsItemSelected(item);  
 }  
  
 @SuppressWarnings(**"StatementWithEmptyBody"**)  
 @Override  
 **public boolean** onNavigationItemSelected(MenuItem item) {  
 *// Handle navigation view item clicks here.* **int** id = item.getItemId();  
  
  
  
 **int** patientId = (**selectedPatient** != **null**) ? **selectedPatient**.getId() : 0;  
 **switch** (id) {  
  
 **case** R.id.***nav\_medications***:  
 loadFragment(**new** ListMedicationFragment(), patientId, **false** );  
 **break**;  
 **case** R.id.***nav\_immunizations***:  
 loadFragment(**new** ListImmunizationFragment(), patientId, **false** );  
 **break**;  
 **case** R.id.***nav\_hospitalizations***:  
 loadFragment(**new** ListHospitalizationFragment(), patientId, **false** );  
 **break**;  
 **case** R.id.***nav\_allergies***:  
 loadFragment(**new** ListAllergyFragment(), patientId, **false**);  
 **break**;  
 **case** R.id.***nav\_office\_visits***:  
 loadFragment(**new** ListOfficeVisitFragment(), patientId, **false** );  
 **break**;  
*//* }  
  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 drawer.closeDrawer(GravityCompat.***START***);  
 **return true**;  
  
 }  
  
  
  
  
 **public void** openRecord(View view) {  
 FragmentTransaction ft = getSupportFragmentManager().beginTransaction();  
 ft.replace(R.id.***content\_panel***, **new** MedicationFragment());  
 ft.commit();  
  
 }  
  
 **public void** editRecord(View view) {  
 *//****TODO will need a checkbox or list view that highlights one selected*** }  
  
  
 **public void** cancelReturntoNavigation(View view) {  
 *//****TODO returns user to Navigation menu*** }  
  
 **public void** cancelRecordActivity(View view) {  
 *//****TODO return user to list view without adding or editing a record*** }  
  
 **public void** loadFragment(Fragment fragment, **int** patientId, **boolean** addToBackstack) {  
  
 FragmentManager fragmentManager = getSupportFragmentManager();  
 Bundle args = **new** Bundle(); *//how to pass arguments* args.putInt(***PATIENT\_ID\_KEY***, patientId);  
  
 fragment.setArguments(args);  
 FragmentTransaction transaction = fragmentManager.beginTransaction()  
 .replace(R.id.***content\_panel***, fragment);  
 **if** (addToBackstack){  
 transaction.addToBackStack(fragment.getClass().getSimpleName());  
 }  
 transaction.commit();  
 }