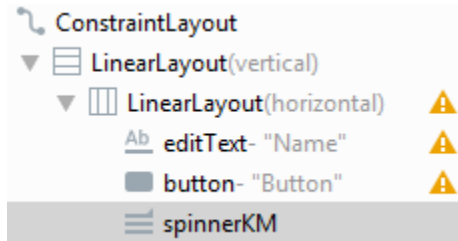


Bước 1: Tạo Empty Project**Bước 2: Tạo giao diện với cấu trúc như sau:****Thêm spinner:**

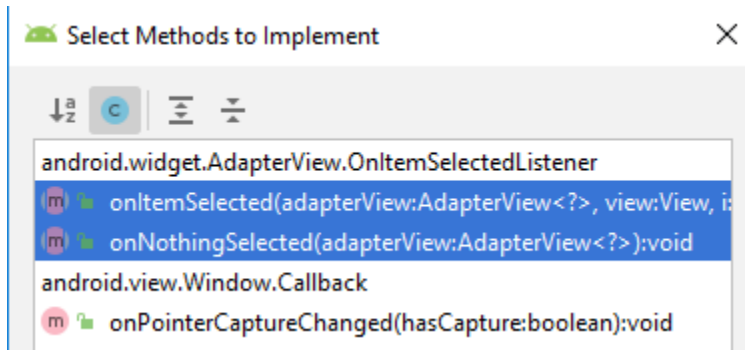
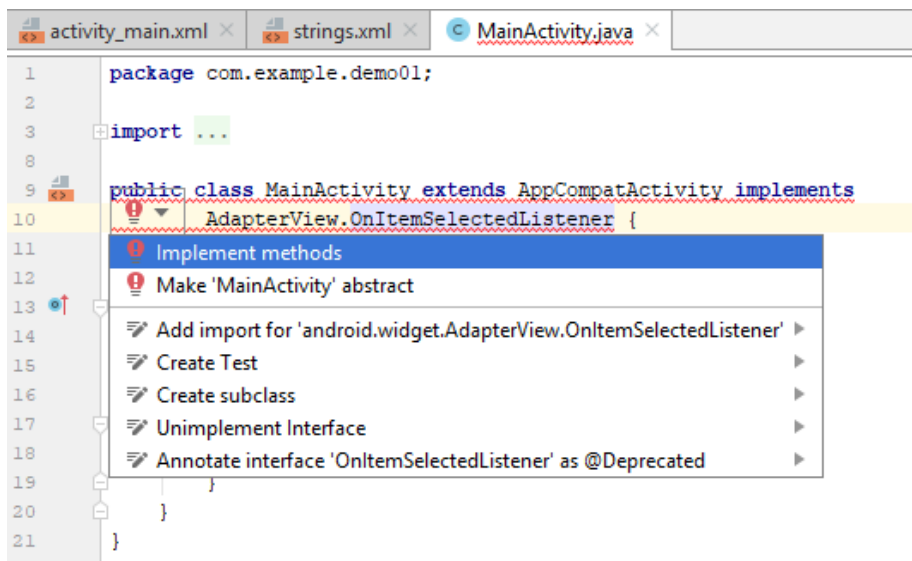
- Tạo danh sách lựa chọn trong strings.xml

```
<string-array name="labels_array">
    <item>Khuyến mãi</item>
    <item>Không KM</item>
</string-array>
```

- Thêm Spinner vào giao diện (alt+enter cho class chưa có)

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    Spinner spinner = findViewById(R.id.spinnerKM);
    if (spinner != null) {
        spinner.setOnItemSelectedListener(this);
    }
}
```

- Thêm Interface cho trình lắng nghe sự kiện chọn Item



- Thêm danh sách vào Spinner

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    Spinner spinner = findViewById(R.id.spinnerKM);
    ArrayAdapter<CharSequence> adapter = ArrayAdapter.createFromResource( context: this,
        R.array.labels_array, android.R.layout.simple_spinner_item);
    // Specify the layout to use when the list of choices appears.
    adapter.setDropDownViewResource
        (android.R.layout.simple_spinner_dropdown_item);
    if (spinner != null) {
        spinner.setOnItemSelectedListener(this);
        spinner.setAdapter(adapter);
    }
}

```

- Lấy giá trị chọn

```

@Override
public void onItemSelected(AdapterView<?> adapterView, View view, int i, long l) {
    String s = adapterView.getItemAtPosition(i).toString();
    Toast.makeText(context MainActivity.this, s, Toast.LENGTH_LONG).show();
}

@Override
public void onNothingSelected(AdapterView<?> adapterView) {

}

```

Bước 3: Thêm RecyclerView vào giao diện

1. Tạo giao diện
2. Tạo Adapter để kết nối
3. Tạo Holder nằm trong Adapter để binding

```

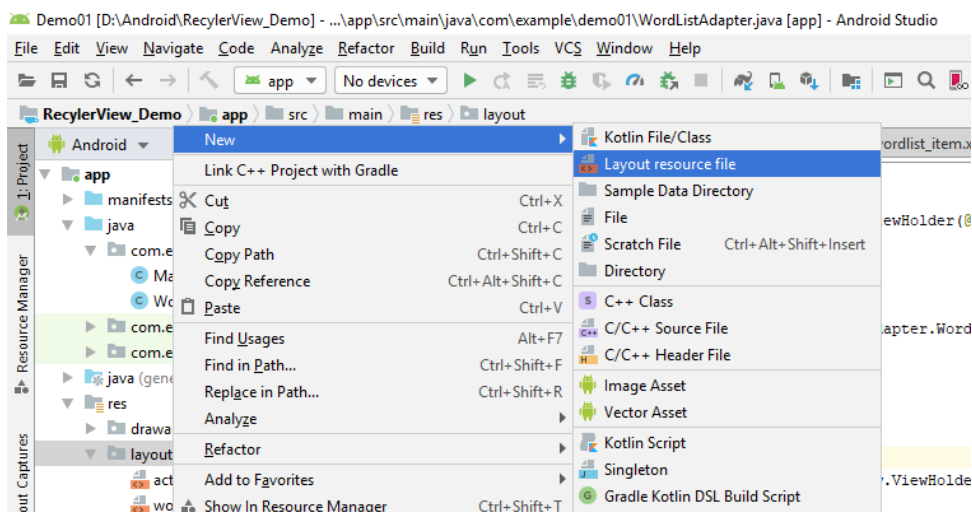
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <LinearLayout...>

    <androidx.recyclerview.widget.RecyclerView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        app:layout_constraintTop_toTopOf="parent" />
</LinearLayout>

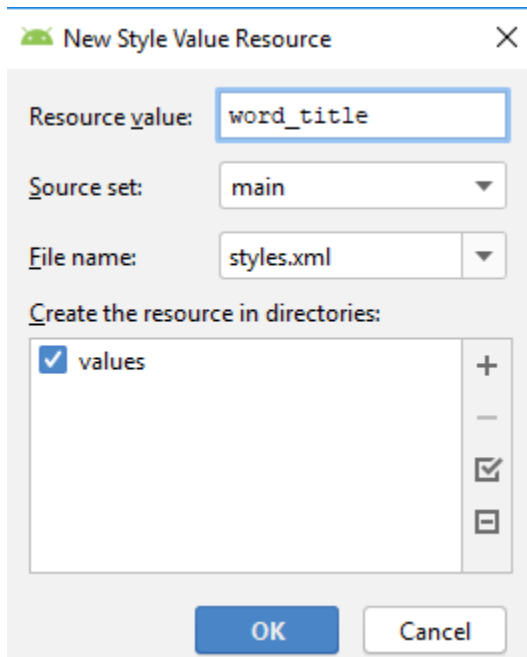
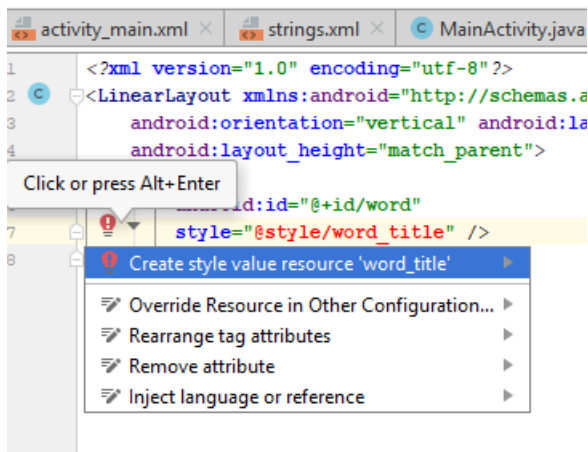
```

- Tạo giao diện cho mỗi item trong RecyclerView (thêm tập tin wordlist_item.xml)

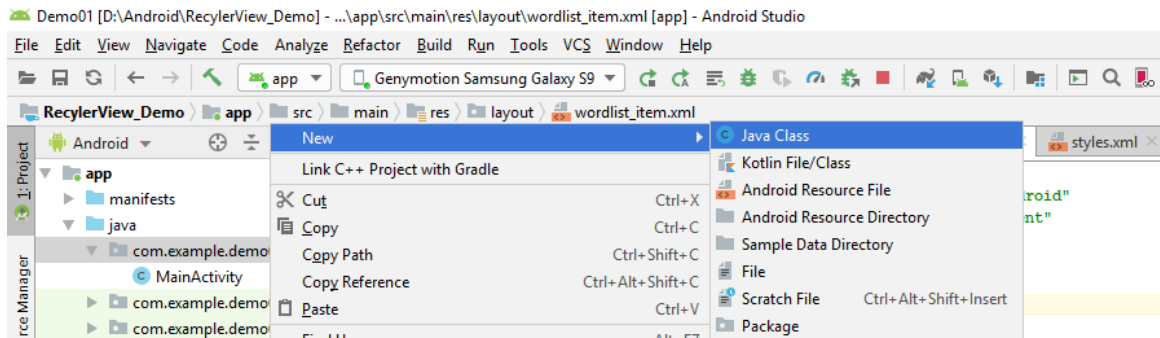


```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TextView
        android:id="@+id/word"
        style="@style/word_title" />
</LinearLayout>
```

- Thêm Resource value



- Tạo Adapter để đưa dữ liệu lên RecyclerView

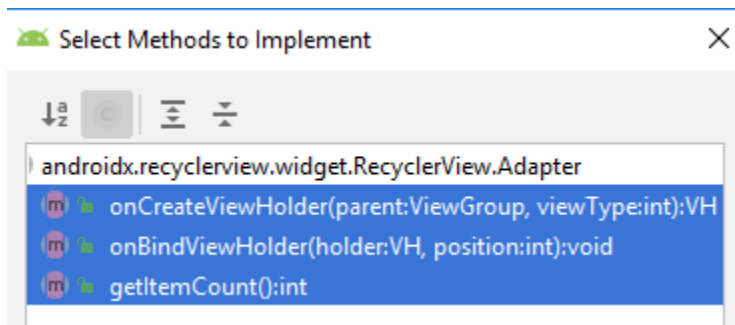
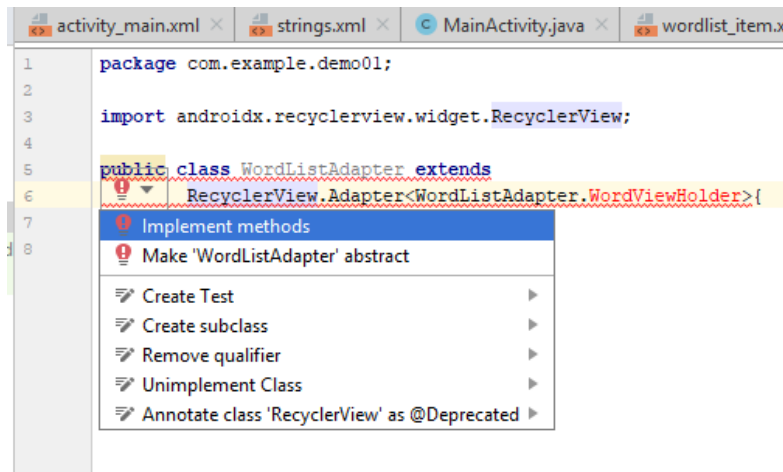


Tạo class WordListAdapter kế thừa từ RecyclerView.Adapter

```
package com.example.demo01;

public class WordListAdapter extends
    RecyclerView.Adapter<WordListAdapter.WordViewHolder>{
}
```

Nhấn Alt_Enter để thêm Package chứa class RecyclerView, tạo ra các phương thức cho class WordViewHolder (là class nằm trong class WordListAdapter)



Tạo Inner Class WordViewHolder, class này kế thừa từ RecyclerView.ViewHolder (là class sử dụng để binding dữ liệu cho các item của RecyclerView). Sau đó viết phương thức khởi tạo cho

class này (Alt+Enter khi import package cho class View). Phương thức khởi tạo này có tham số truyền vào là 1 View (là 1 control TextView chứa thông tin 1 item của RecyclerView và WordListAdapter mà nó sử dụng để liên kết.

```

1 package com.example.demo01;
2
3 import android.view.ViewGroup;
4
5 import androidx.annotation.NonNull;
6 import androidx.recyclerview.widget.RecyclerView;
7
8 public class WordListAdapter extends
9     RecyclerView.Adapter<WordListAdapter.WordViewHolder>{
10
11     // Create class 'WordViewHolder'
12     // Create enum 'WordViewHolder'
13     // Create interface 'WordViewHolder'
14     // Create Test
15     // Create subclass
16     // Unimplement Class
17     public void onBindViewHolder(@NonNull WordListAdapter.WordViewHolder holder, int position) {
18
19     }
20
21     @Override
22     public int getItemCount() {
23         return 0;
24     }
25 }

```

```

public class WordViewHolder extends RecyclerView.ViewHolder{
    public WordViewHolder(View itemView, WordListAdapter adapter) {
        super(itemView);
    }
}

```

Thêm thuộc tính cho WordViewHolder để chứa thông tin TextView cho item và thuộc tính thể hiện WordListAdapter mà WordViewHolder thuộc về. Gán giá trị cho chúng trong phương thức khởi tạo của WordViewHolder. (Alt+Enter khi cần thiết)

```

public class WordViewHolder extends RecyclerView.ViewHolder{
    public final TextView wordItemView;
    final WordListAdapter mAdapter;
    public WordViewHolder(View itemView, WordListAdapter adapter) {
        super(itemView);
        wordItemView = itemView.findViewById(R.id.word);
        this.mAdapter = adapter;
    }
}

```

Thêm thuộc tính chứa danh sách dữ liệu cho WordListAdapter (Alt+Enter nếu cần thiết) và viết hàm khởi tạo cho WordListAdapter truyền danh sách từ ngoài vào để khởi tạo. Ngoài ra, phương thức khởi tạo còn truyền vào đối tượng Context (là cầu nối giữa các thành phần, sử dụng để giao tiếp giữa các thành phần, khởi tạo các thành phần và truy cập các thành phần như Activity, Content Provider, BroadcastReceiver hoặc truy cập vào các resources và filesystems) ở đây là để truy cập đến Activity bên ngoài đang chứa RecyclerView. Thuộc tính kiểu LayoutInflater (là 1 component giúp chuyển layout file(Xml) thành View(Java code) trong Android) cũng được thêm vào để truy xuất giao diện từ Activity bên ngoài.

```
public class WordListAdapter extends
    RecyclerView.Adapter<WordListAdapter.WordViewHolder>{
    private final LinkedList<String> mWordList;
    private LayoutInflater mInflater;
    public WordListAdapter(Context context,
        LinkedList<String> wordList) {
        mInflater = LayoutInflater.from(context);
        this.mWordList = wordList;
    }
}
```

Viết lại phương thức onCreateViewHolder để tạo Holder chứa dữ liệu vào RecyclerView

```
@Override
public WordListAdapter.WordViewHolder onCreateViewHolder(@NonNull Vi
    View itemView = mInflater.inflate(R.layout.wordlist_item,
        parent, attachToRoot: false);
    return new WordViewHolder(itemView, adapter: this);
}
```

Phương thức inflate để chuyển đổi 1 xml layout file thành 1 View trong java

View view = layoutInflater.inflate(int resource, ViewGroup parent)

View view = layoutInflater.inflate(int resource, ViewGroup parent, boolean attachToRoot)

☞ **Tham số thứ nhất là:** int resource, nó chính là xml layout file muốn chuyển đổi thành View.

☞ **Tham số thứ hai là:** ViewGroup parent, là ViewGroup nơi mà xml layout file(tham số thứ nhất) có thể được nhúng vào, LayoutInflater sẽ chuyển đổi xml layout file thành View và sử dụng các thuộc tính phù hợp với ViewGroup parent.

☞ **Tham số thứ ba là:** attachToRoot, khi mà attachToRoot=true thì ngay sau khi quá trình chuyển đổi xml file(resource) thành View hoàn thành thì nó sẽ nhúng View đó vào ViewGroup

parent (RIGHT NOW) , khi attachToRoot = false thì nó chỉ chuyển đổi xml file(resource) thành View trong java mà không thêm ngay vào ViewGroup(NOT NOW).

Viết lại phương thức onBindViewHolder để binding dữ liệu cho RecyclerView

```
@Override
public void onBindViewHolder(@NonNull WordListAdapter.WordVi
    String mCurrent = mWordList.get(position);
    holder.wordItemView.setText(mCurrent);
}
```

Viết lại phương thức trả về số Item cho Adapter

```
@Override
public int getItemCount() { return mWordList.size(); }
```

Bước 4: Sử dụng RecyclerView load dữ liệu

- Tạo thuộc tính mWordList chứa danh sách dữ liệu cần hiển thị trên RecyclerView, tạo dữ liệu hiển thị

```
private final LinkedList<String> mWordList = new LinkedList<>();
private RecyclerView mRecyclerView;
private WordListAdapter mAdapter;
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    for (int i = 1; i < 20; i++) {
        mWordList.addLast("Word " + i);
    }
}
```

- Tạo thuộc tính chứa RecyclerView và Adapter trong MainActivity

```
public class MainActivity extends AppCompatActivity implements
    AdapterView.OnItemClickListener {
    private final LinkedList<String> mWordList = new LinkedList<>();
    private RecyclerView mRecyclerView;
    private WordListAdapter mAdapter;
```

- Truy xuất RecyclerView sử dụng Adapter để binding dữ liệu


```

    if (spinner != null) {
        spinner.setOnItemSelectedListener(this);
        spinner.setAdapter(adapter);
    }
    // Get a handle to the RecyclerView.
    mRecyclerView = findViewById(R.id.recycler);
    // Create an adapter and supply the data to be displayed.
    mAdapter = new WordListAdapter( context: this, mWordList);
    // Connect the adapter with the RecyclerView.
    mRecyclerView.setAdapter(mAdapter);
    // Give the RecyclerView a default layout manager.
    mRecyclerView.setLayoutManager(new LinearLayoutManager( context: this));
}

```

Chạy ứng dụng, bị lỗi không hiển thị được do phần thiết kế giao diện 1 item của RecyclerView (Thêm layout_width, layout_height cho TextView) và giao diện MainActivity (Chỉnh thuộc tính layout_height các thành phần là wrap_content)

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <TextView
        android:id="@+id/word"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        style="@style/word_title"/>
</LinearLayout>

```

Bước 5: Xử lý lọc theo từ khóa nhập vào EditText, viết phương thức cho nút nhấn

- Tạo danh sách mới (danh sách kết quả)
- Duyệt danh sách dữ liệu gốc, kiểm tra điều kiện theo từ khóa, thêm vào danh sách kết quả
- Hiển thị lại trên RecyclerView (tạo mới Adapter với dữ liệu mới, hiển thị lại dữ liệu)

```
public void search_by_name(View v) {  
    LinkedList<String> result = new LinkedList<>();  
    int count=mWordList.size();  
    EditText txt = (EditText)findViewById(R.id.editText);  
    for (int i = 0; i <count ; i++) {  
        if(mWordList.get(i).indexOf(txt.getText().toString())>=0)  
        {  
            result.add(mWordList.get(i));  
        }  
    }  
    mAdapter = new WordListAdapter( context: this, result);  
    mRecyclerView.setAdapter(mAdapter);  
    mRecyclerView.setLayoutManager(new LinearLayoutManager( context: this));  
}
```

Chạy kiểm tra kết quả

