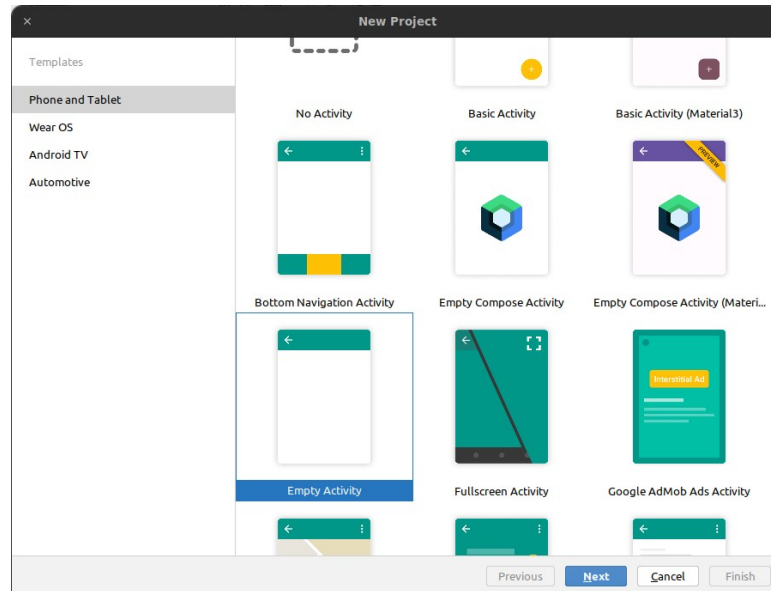


4ο Μάθημα.

Walkthrough: Εισαγωγή στο Αντικειμενοστρεφές Mobile Development

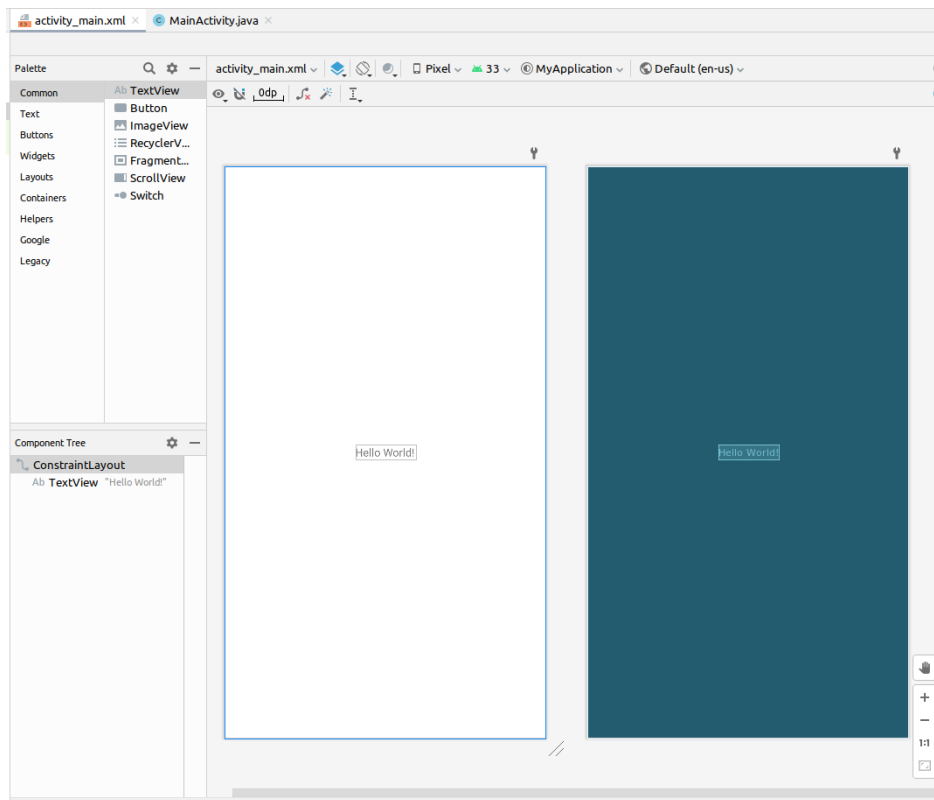
Θα δημιουργήσουμε μία κενή δραστηριότητα. Επιλέγουμε New Project, έπειτα Empty Activity και πατάμε **Next**.



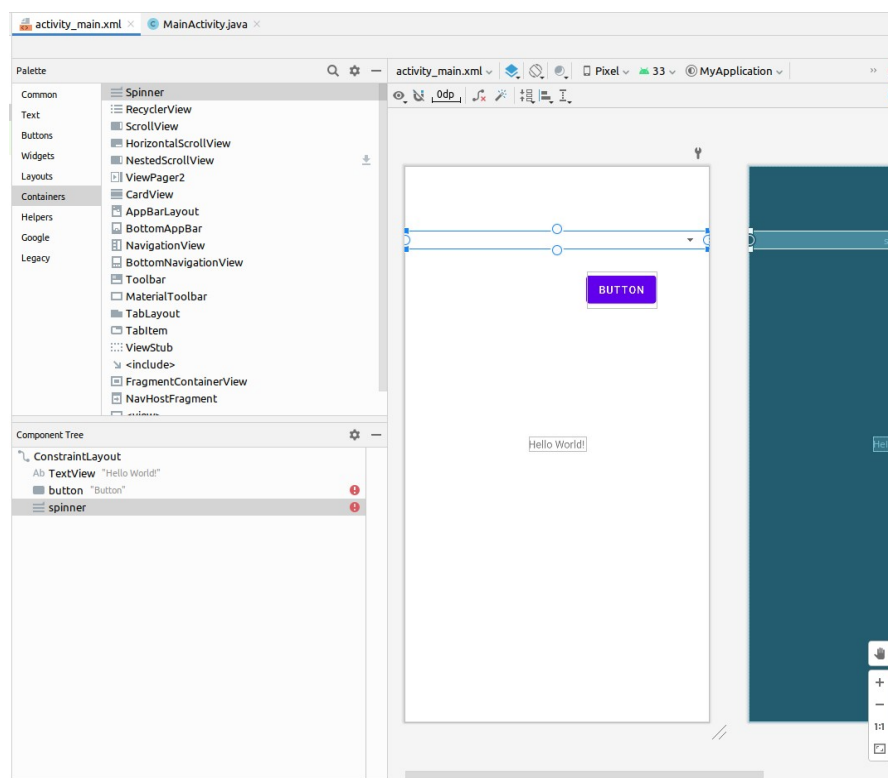
Στη επόμενη οθόνη επιλέγουμε τα εξής:

Το Save location δεν χρειάζεται να είναι το ίδιο με παραπάνω. Επιλέγουμε **Finish**.

Αφού ολοκληρωθεί το sync από το Gradle, ανοίγουμε το αρχείο activity_main.xml.



Από την παλέτα πάνω αριστερά, και την επιλογή common με drag & drop τοποθετούμε ένα κουμπί πάνω στο layout.



Από την επιλογή Containers στην παλέτα επιλέγουμε Spinner (1ο στη σχετική λίστα) και το τοποθετούμε στο layout.

Κάνουμε τις παρακάτω αλλαγές στο spinner:

id: cars

layout_width: wrap_content

layout_height: wrap_content

minheight:48dp

Θα προσθέσουμε τις επιλογές του spinner στατικά. Ανοίγουμε το strings.xml και προσθέτουμε τα παρακάτω:

```
<string-array name="brands">
    <item>Tomota</item>
    <item>WW</item>
    <item>Fissan</item>
</string-array>
```

ώστε αυτά να βρίσκονται εντός των resources.

Από τα attributes στο spinner αλλάζουμε το entries. Κάνουμε κλικ στα δεξιά του πεδίου και από το μενού που εμφανίζεται επιλέγουμε brands που είναι και το όνομα του array που μόλις δημιουργήσαμε.



Κάνουμε τις παρακάτω αλλαγές στο κουμπί:

id: pick_car

text: Pick a brand (το οποίο και κάνουμε extract στο string.xml)

Κάνουμε τις παρακάτω αλλαγές στο textview:

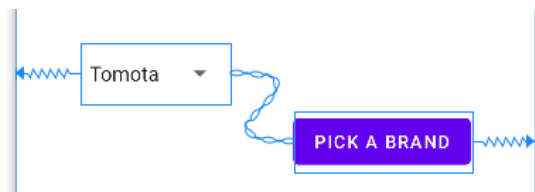
id:models

text: το αφήνουμε κενό

Δημιουργία αλυσίδων και ορισμός περιορισμών

Από το component tree με το πλήκτρο shift πατημένο επιλέγουμε και το pick_car και το cars

Στη συνέχεια κάνουμε δεξί κλικ στα επιλεγμένα και επιλέγουμε Chains- → Create Horizontal Chain

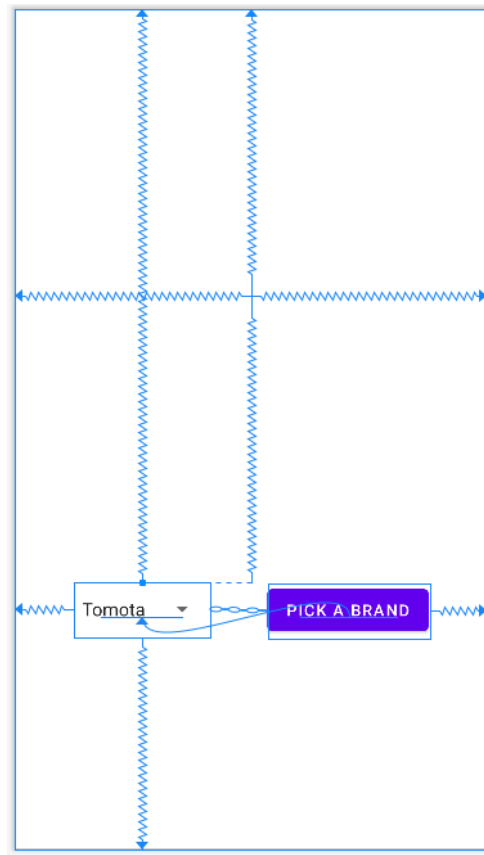


Βλέπουμε ότι έχει δημιουργηθεί μία οριζόντια αλυσίδα που συνδέει τα δύο Views, καθώς και constraints προς το πλαίσιο από την πλευρά του κάθε View.

Κάνουμε δεξί κλικ έχοντας επιλεγμένα και τα δύο Views και επιλέγουμε Align- → Baselines ώστε να τα ευθυγραμμίσουμε.

Στη συνέχεια, περιορίζουμε το spinner ώστε να είναι κάτω από το πάνω πλαίσιο και πάνω από το κάτω πλαίσιο. Ορίζουμε το κατακόρυφο bias σε 70.

Ορίζουμε τον κάτω περιορισμό του Textview να είναι πάνω από το Spinner.



Θα προσθέσουμε λειτουργικότητα στο κουμπί

Στο `mainActivity.java` (εντός της κλάσης) δημιουργούμε μία μέθοδο ως εξής:

```
public void myOnClickPickBrand(View v){
}
```

Είναι σημαντικό η μέθοδος να είναι `public void` και να έχει ως παράμετρο ένα `view` ώστε να μπορέσει να χρησιμοποιηθεί από το `ide`. Εδώ βλέπουμε έναν άλλον τρόπο να προσθέσουμε λειτουργικότητα σε ένα κουμπί.

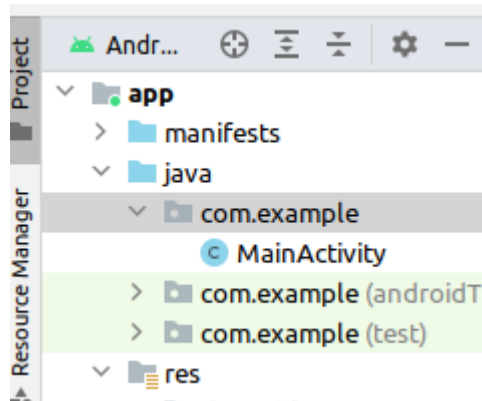
Φυσικά, μπορούμε να το υλοποιήσουμε και προγραμματιστικά, χρησιμοποιώντας `findViewById` και υλοποιώντας έναν `listener`.

Επιστρέφοντας στο `activity_main.xml`, επιλέγουμε το κουμπί και αλλάζουμε το attribute `onClick`, το οποίο πλέον, όπως μπορούμε να παρατηρήσουμε έχει μία επιλογή. Τη μέθοδο `myOnClickPickBrand`, την οποία και υλοποιήσαμε πριν λίγο. Την ορίζουμε στο `onClick`.

keyboardNavigationCluster	-	0
onClick	myOnClickPickBrand	▼ 0
toggleCheckedStateOnClick	-	0

Τώρα, με βάση την επιλογή στο `spinner`, όταν πατηθεί το κουμπί θα εμφανίζεται διαφορετικό κείμενο στο `TextView`

Για να το επιτύχουμε αυτό θα δημιουργήσουμε σε πρώτη φάση μία νέα κλάση.
Από το Project, επιλέγουμε το πακέτο com.example



κάνουμε δεξί κλικ → New- → Java Class.

Στο πεδίο διαλόγου που εμφανίζεται, εισάγουμε CarPicker που είναι και το όνομα της κλάσης και συμπληρώνουμε τον παρακάτω κώδικα ώστε να έχουμε αυτό το αποτέλεσμα:

```
package com.example;
```

```
import java.util.ArrayList;
```

```
import java.util.List;
```

```
public class CarPicker {  
    List<String> getCars(String s){  
        List<String> models = new ArrayList<String>();  
        switch (s){  
            case "Tomota":  
                models.add("Kauris");  
                models.add("Yamis");  
                models.add("Gav14");  
                break;  
            case "WW":  
                models.add("Molf");  
                models.add("Solo");  
                break;  
            default:  
                break;  
        }  
        return models;  
    }  
}
```

Στο MainActivity.java δημιουργώ ως πεδίο ένα αντικείμενο της κλάσης που δημιούργησα:

```
public class MainActivity extends AppCompatActivity {
```

```
    private CarPicker co = new CarPicker();
```

Στη συνέχεια, θα προσθέσω λειτουργικότητα στη μέθοδο myOnClickPickBrand,
Προσθέτω τον παρακάτω κώδικά, ώστε η myOnClickPickBrand να αλλάζει το textfield

```
public void myOnClickPickBrand(View v) {  
    TextView brands = findViewById(R.id.models);  
    Spinner color = findViewById(R.id.cars);  
    String carMake = String.valueOf(color.getSelectedItem());
```

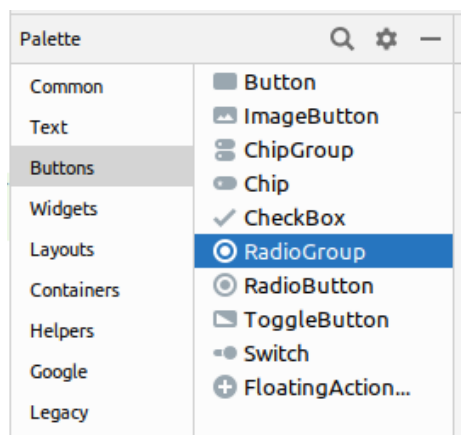
```

List<String> brandsList = cp.getCars(carMake);
if (brandsList.size() == 0) {
    brands.setText("There are no models for this brand. Pick another!");
} else {
    StringBuilder brandsFormatted = new StringBuilder();
    for (String brand : brandsList)
        brandsFormatted.append(brand).append("\n");
    brands.setText(brandsFormatted);
}
}

```

Επόμενο βήμα μας είναι να δημιουργήσουμε δυναμικά μία λίστα από radio buttons. Κάθε φορά που αλλάζουμε επιλογή και πατάμε το κουμπί, θα εμφανίζεται και διαφορετική λίστα. Με την επιλογή radio button θα εμφανίζεται ένα Toast.

Αφαιρούμε το Textview και στη θέση του βάζουμε ένα radiogroup. Το στοιχίζουμε με τον ίδιο τρόπο. Του δίνουμε id car_radio.



Δημιουργώ δύο νέες κλάσεις.

1. Brand

```
package com.example;
```

```
import java.util.ArrayList;
```

```
import java.util.List;
```

```
public class Brand {
```

```
    private String name;
```

```
    private ArrayList<String> mList = new ArrayList<String>();
```

```
    public Brand(String n) {
```

```
        name = n;
```

```
    }
```

```
    public boolean isBrand(String brand) {
```

```
        return name.equals(brand);
```

```
    }
```

```
    public void addModel(String model) {
```

```

    mList.add(model);
}

public List<String> getModels() {
    return mList;
}
}

```

2. CarBrandsList

```

public class CarBrandsList {
    ArrayList<Brand> bList = new ArrayList<Brand>();

    public CarBrandsList() {

    }

    public void addModel(String brand, String model) {
        for (int i=0; i<bList.size(); i++) {
            if (bList.get(i).isBrand(brand)) {
                bList.get(i).addModel(model);
            }
        }
    }

    public List<String> getCars(String brand) {
        for (int i=0; i<bList.size(); i++) {
            if (bList.get(i).isBrand(brand)) {
                return bList.get(i).getModels();
            }
        }
        return null;
    }

    public void addBrand(Brand b) {
        bList.add(b);
    }
}

```

Τροποποιούμε το strings.xml

```

<resources>
    <string name="app_name">My Application</string>
    <string name="pick_a_brand">Pick a brand</string>
    <string name="Tomota" translatable="false">Tomota</string>
    <string name="WW" translatable="false">WW</string>
    <string name="Fissan" translatable="false">Fissan</string>

    <string-array name="brands">
        <item>@string/Tomota</item>
        <item>@string/WW</item>
        <item>@string/Fissan</item>
    </string-array>
    <string name="toast_no_cars">%s has no models. Pick another!</string>
</resources>

```

Τροποποιούμε το MainActivity.java ώστε να είναι ως εξής:

```
public class MainActivity extends AppCompatActivity {

    private CarPicker cp = new CarPicker();
    private CarBrandsList cbList;
    RadioGroup rg;

    private void createRadioButtons(RadioGroup rg, List<String> brandsList) {
        int size = brandsList.size();
        final RadioButton[] rb = new RadioButton[size];
        for(int i=0; i<size; i++){
            rb[i] = new RadioButton(this);
            rb[i].setText(" " + brandsList.get(i));
            rb[i].setId(i + 100);
            rg.addView(rb[i]);
        }
    }

    public void myOnClickPickBrand(View v) {

        Spinner dropDown = findViewById(R.id.cars);
        String brand = String.valueOf(dropDown.getSelectedItem());

        List<String> brandsList = cbList.getCars(brand);
        rg.removeAllViews();

        if(brandsList.size() == 0){
            Toast.makeText(getApplicationContext(), getString(R.string.toast_no_cars, brand),
            Toast.LENGTH_SHORT).show();
        } else{
            rg.setOrientation(RadioGroup.VERTICAL); //or RadioGroup.VERTICAL
            createRadioButtons(rg, brandsList);
            rg.setOnCheckedChangeListener(new RadioGroup.OnCheckedChangeListener() {
                @Override
                public void onCheckedChanged(RadioGroup group, int checkedId) {
                    RadioButton rb= findViewById(checkedId);
                    Toast.makeText(getApplicationContext(), brand+" "+rb.getText(),
                    Toast.LENGTH_SHORT).show();
                }
            });
        }
    }

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        cbList = new CarBrandsList();
        cbList.addBrand(new Brand(getString(R.string.Tomota)));
        cbList.addBrand(new Brand(getString(R.string.WW)));
    }
}
```



```
cbList.addBrand(new Brand(getString(R.string.Fissan))));
```

```
cbList.addModel(getString(R.string.Tomota), "Kauris");
```

```
cbList.addModel(getString(R.string.Tomota), "Yamis");
```

```
cbList.addModel(getString(R.string.Tomota), "Gav14");
```

```
cbList.addModel(getString(R.string.WW), "Molf");
```

```
cbList.addModel(getString(R.string.WW), "Solo");
```

```
rg = findViewById(R.id.car_radio);
```

```
}
```

```
}
```