

Nhà phát triển Android

Khóa học cơ bản (Phiên bản 2)



Cập nhật lần cuối: Tue Sep 11, 2018

Khóa học này được tạo bởi nhóm Đào tạo nhà phát triển của Google.

- Để biết chi tiết về khóa học, bao gồm các liên kết đến tất cả các chương khái niệm, ứng dụng và trang trình bày, hãy xem

[Kiến thức cơ bản về nhà phát triển Android \(Phiên bản 2\).](#)

developer.android.com/courses/ a dfv2

Lưu ý: Khóa học này sử dụng các thuật ngữ "codelab" và "thực hành" thay thế cho nhau.

Chúng tôi khuyên bạn nên sử dụng phiên bản o nline của khóa học này thay vì PDF tĩnh này để đảm bảo bạn đang sử dụng nội dung mới nhất.

Xem [d eveloper.android.com/courses/adf-v2](https://developer.android.com/courses/adf-v2).

Bài 2: Trải nghiệm người dùng

PDF này chứa ảnh chụp nhanh một lần về các bài học trong **Unit 2: Trải nghiệm người dùng**.

Bài học trong bài học này

Bài 4: Tương tác người dùng

4.1 : Hình ảnh có thể nhấp

4.2: Điều khiển đầu vào

4.3 : Menu và bộ chọn

4.4 : Điều hướng người dùng

4.5: Người tái chế View

Bài 5: Trải nghiệm người dùng thú vị

5.1 : Có thể vẽ, kiểu và chủ đề

5.2 : Thẻ và màu sắc

5.3 : Bố cục thích ứng

Bài học 6: Kiểm tra giao diện người dùng

6.1 : Espresso để kiểm tra giao diện người dùng

Hình ảnh có thể nhấp

Giới thiệu

Giao diện người dùng (UI) xuất hiện trên màn hình của thiết bị chạy Android bao gồm một hệ thống phân cấp các đối tượng được gọi là *views*. Mọi yếu tố của màn hình đều là một chữ [View](#).

Lớp `View` đại diện cho khối xây dựng cơ bản cho tất cả các thành phần giao diện người dùng. `View` là lớp cơ sở cho các lớp cung cấp các thành phần giao diện người dùng tương tác, chẳng hạn như [các phần tử Button](#). `Button` là một yếu tố giao diện người dùng mà người dùng có thể nhấn hoặc nhấp để thực hiện một hành động.

Bạn có thể biến bất kỳ `View` nào, chẳng hạn như [ImageView](#), thành một phần tử giao diện người dùng có thể chạm hoặc nhấp vào. Bạn phải lưu trữ hình ảnh cho `ImageView` trong thư mục `drawables` của dự án của bạn.

Trong thực tế này, bạn học cách sử dụng hình ảnh làm yếu tố mà người dùng có thể nhấn hoặc nhấp vào.

Những điều bạn nên biết

Bạn sẽ có thể:

- Tạo một dự án Android Studio từ một mẫu và tạo bố cục chính.
- Chạy ứng dụng trên trình mô phỏng hoặc thiết bị được kết nối.
- Tạo và chỉnh sửa các thành phần giao diện người dùng bằng trình chỉnh sửa bố cục và mã XML.
- Truy cập các thành phần giao diện người dùng từ mã của bạn bằng [findViewById\(\)](#).
- Xử lý một cú [nhấp chuột](#) `Button`.
- Hiện thị thông [báo Toast](#).
- Thêm hình ảnh vào thư mục có thể truy cập của dự án.

Những gì bạn sẽ học

- Cách sử dụng hình ảnh làm yếu tố tương tác để thực hiện một hành động.
- Cách đặt thuộc tính cho các phần tử ImageView trong trình chỉnh sửa bố cục.
- Cách thêm một onClick() method để hiển thị thông báo Toast.

Bạn sẽ làm gì

- Tạo một dự án Android Studio mới cho một ứng dụng đặt món tráng miệng giả sử dụng hình ảnh làm yếu tố tương tác.
- Đặt các trình xử lý onClick() cho các hình ảnh để hiển thị các thông báo Toast khác nhau.
- Thay đổi nút hành động nổi do mẫu cung cấp để nó hiển thị một biểu tượng khác và khởi chạy một biểu tượng A khác.

Tổng quan về ứng dụng

Trong thực tế này, bạn tạo và xây dựng một ứng dụng mới bắt đầu với mẫu Hoạt động cơ bản bắt chước ứng dụng đặt món tráng miệng. Người dùng có thể nhấn vào một hình ảnh để thực hiện một hành động — trong trường hợp này hiển thị thông báo Toast — như thể hiện trong hình bên dưới. Người dùng cũng có thể nhấn vào nút giỏ hàng để chuyển sang chương trình A tiếp theo.



Nhiệm vụ 1: Thêm hình ảnh vào bố cục

Bạn có thể làm cho một chế độ xem có thể nhấp được, như một nút, bằng cách thêm thuộc tính `android:onClick` trong bố cục XML. Ví dụ: bạn có thể làm cho một hình ảnh hoạt động giống như một nút bằng cách thêm `android:onClick` vào

[Chế độ xem hình ảnh.](#)

Trong nhiệm vụ này, bạn tạo một nguyên mẫu của một ứng dụng để đặt món tráng miệng từ một quán cà phê. Sau khi bắt đầu một dự án mới dựa trên mẫu Hoạt động cơ bản, bạn sửa đổi "Hello World" TextView với văn bản thích hợp và thêm hình ảnh mà người dùng có thể chạm.

1.1 Bắt đầu dự án mới

1. Bắt đầu một dự án Android Studio mới với tên ứng dụng **Droid Cafe**.
2. Chọn mẫu **Hoạt động Basic** và chấp nhận tên Activity mặc định (**MainActivity**). Đảm bảo các tùy chọn **Bộ cục Generate File** và **Backwards Compatibility (AppCompat)** được chọn.
3. Nhấp vào **Finish**.

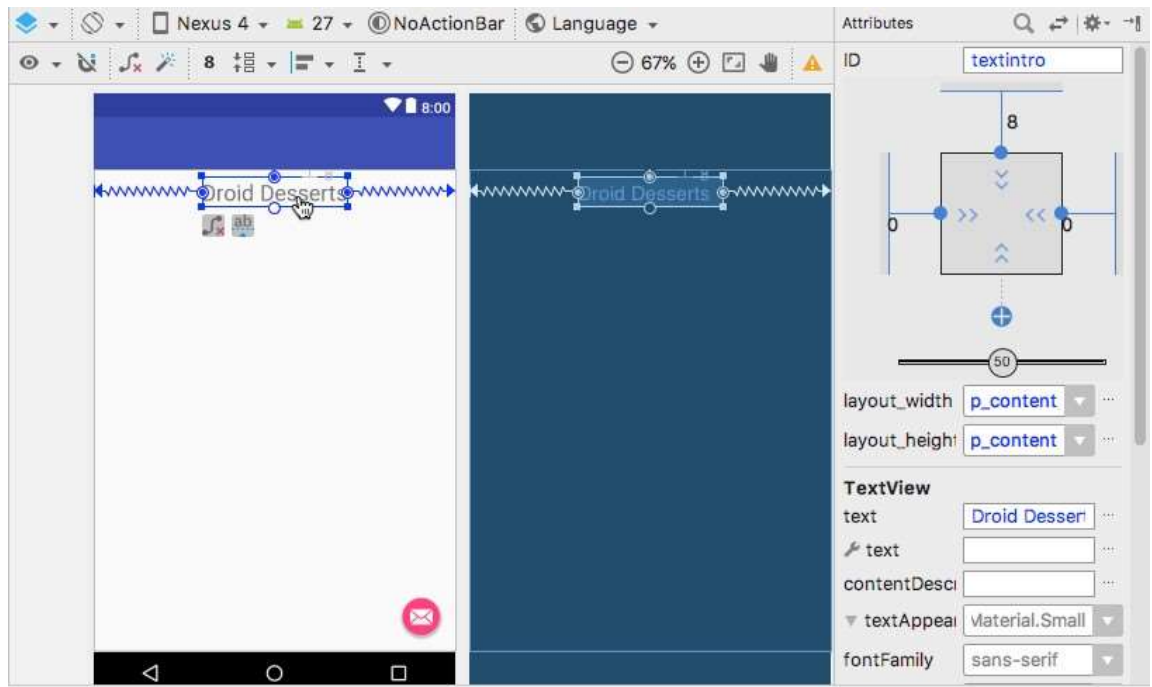
Dự án mở ra với hai bộ cục trong **thư mục** bộ cục **res**: **activity_main.xml** cho thanh ứng dụng và nút hành động nổi (mà bạn không thay đổi trong tác vụ này) và **content_main.xml** cho mọi thứ khác trong bộ cục.

4. Mở **content_main.xml** và nhấp vào tab **Design** (nếu nó chưa được chọn) để hiển thị trình chỉnh sửa bộ cục.
5. Chọn "Hello World" **TextView** trong bộ cục và mở ngăn **Attributes**.
6. Thay đổi các thuộc tính **text** như sau:

Trường thuộc tính	Nhập như sau:
ID	văn bản giới thiệu
Nhắn tin	Thay đổi thế giới Hello thành món tráng miệng Droid
textStyle	B (in đậm)
Kích thước văn bản	24 điểm

Điều này thêm thuộc tính android:id vào TextView với id được đặt thành textintro, thay đổi văn bản, làm cho văn bản in đậm và đặt kích thước văn bản lớn hơn là 24sp.

7. Xóa ràng buộc kéo dài từ dưới cùng của textintro TextView đến cuối bố cục, để TextView gắn vào đầu bố cục và chọn **8** (8dp) cho lề trên cùng như hình dưới đây.



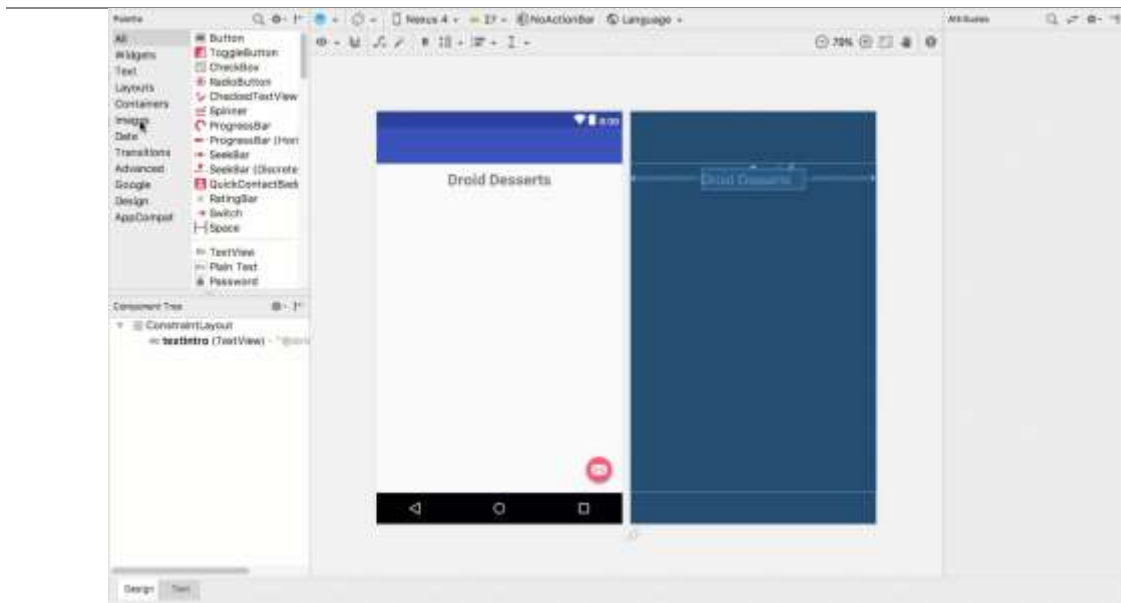
8. Trong bài học trước, bạn đã học cách trích xuất tài nguyên chuỗi từ chuỗi văn bản theo nghĩa đen.
Nhấp vào tab **Text** để chuyển sang mã XML và giải nén chuỗi " Droid Desserts" trong TextView và nhập vào **tro_text** làm tên tài nguyên chuỗi.

Thêm hình ảnh

Ba hình ảnh (donut_circle.png, froyo_circle.png và icecream_circle.png) được cung cấp cho ví dụ này, bạn có thể [tự tải mà bạn có thể tự tải](#). Thay vào đó, bạn có thể thay thế hình ảnh của riêng mình dưới dạng tệp PNG, nhưng chúng phải có kích thước khoảng 113 x 113 pixel để sử dụng trong ví dụ này.

Bước này cũng giới thiệu một kỹ thuật mới trong trình chỉnh sửa bố cục: sử dụng **nút Fix** trong các thông báo cảnh báo để trích xuất tài nguyên chuỗi.

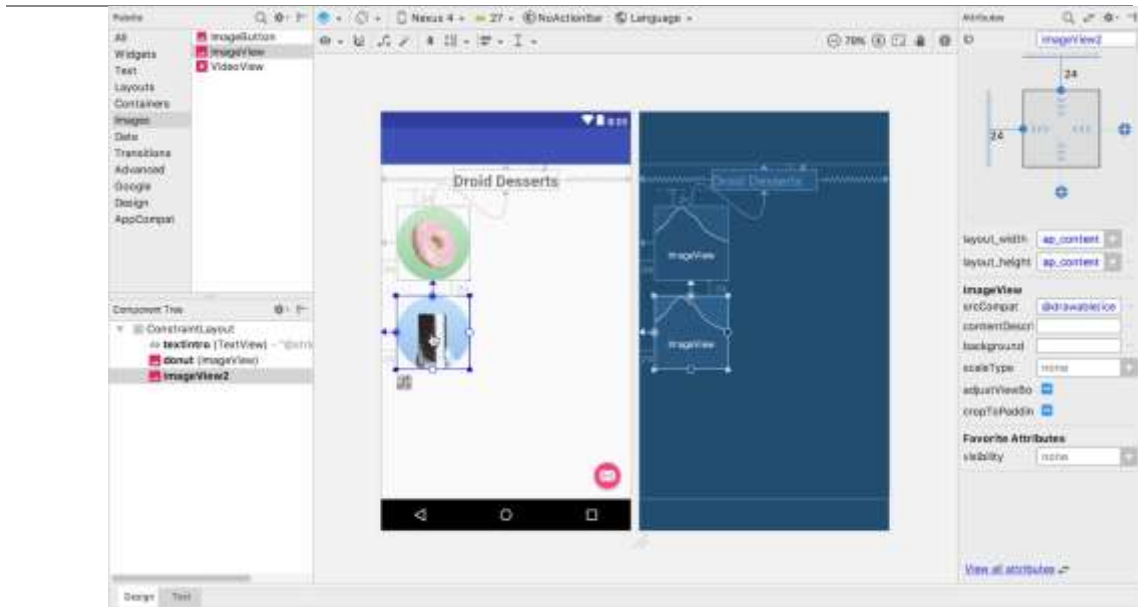
1. Để sao chép hình ảnh vào dự án của bạn, trước tiên hãy đóng dự án.
2. Sao chép các tệp hình ảnh vào thư mục **drawable** của dự án của bạn. Tìm **thư mục drawable** trong một dự án bằng cách sử dụng đường dẫn sau: *project_name ứng dụng > > src > main > res > drawable*.
3. Mở lại dự án của bạn.
4. Mở **tệp content_main.xml** và nhấp vào tab **chữ ký D** (nếu nó chưa được chọn).
5. Kéo ImageView vào bố cục, chọn hình ảnh **donut_circle** cho nó và hạn chế nó ở TextView trên cùng và sang phía bên trái của bố cục với lề 24 (24dp) cho cả hai ràng buộc, như được hiển thị trong hình động bên dưới.



6. Trong ngăn **Attributes**, nhập các giá trị sau cho các thuộc tính:

Trường thuộc tính	Nhập như sau:
ID	Donut
Nội dung mô tả	Bánh rán được tráng men và rắc kẹo. (Bạn có thể sao chép/dán văn bản vào trường.)

7. Kéo **ImageView** thứ hai vào bố cục, chọn **icecream_circle** hình ảnh cho nó và hạn chế nó ở cuối **ImageView** đầu tiên và sang phía bên trái của bố cục với lề **24** (24dp) cho cả hai ràng buộc.



8. Trong ngăn **Attributes**, nhập các giá trị sau cho các thuộc tính:


Trường thuộc tính	Nhập như sau:
ID	ice_cream
nội dungMô tả	Bánh mì kẹp kem có bánh xốp sô cô la và nhân vani. (Bạn có thể sao chép/dán văn bản vào trường.)

9. Kéo ImageView thứ ba vào bố cục, chọn hình ảnh **froyo_circle** cho nó và hạn chế nó ở dưới cùng của ImageView thứ hai và sang phía bên trái của bố cục với lề 24 (24dp) cho cả hai ràng buộc.

10. Trong ngăn **Attributes**, nhập các giá trị sau cho các thuộc tính:

Trường thuộc tính	Nhập như sau:
ID	Froyo

nội dungMô tả	FroYo là sữa chua đông lạnh tự phục vụ cao cấp. (Bạn có thể sao chép/dán văn bản vào trường.)
---------------	---

11. Nhấp vào biểu tượng cảnh báo  ở góc trên bên trái của trình chỉnh sửa bố cục để mở ngăn cảnh báo, ngăn này sẽ hiển thị cảnh báo về văn bản được mã hóa cứng:



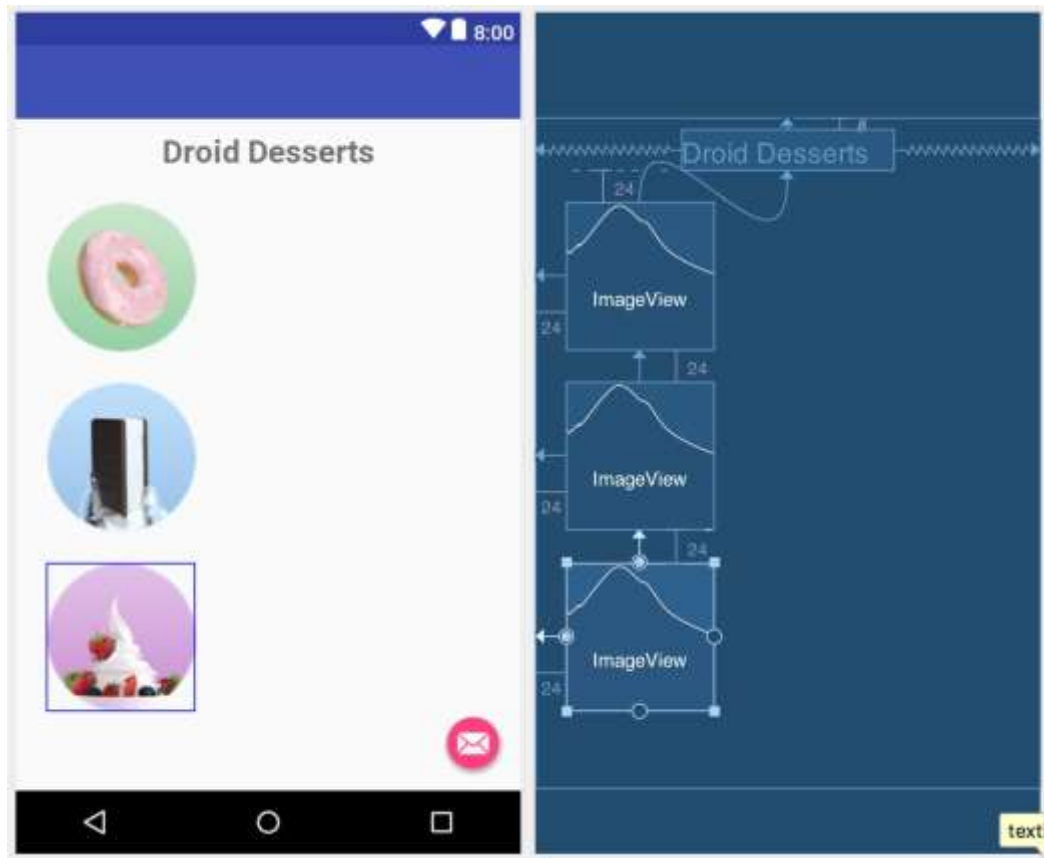
12. Mở rộng từng cảnh báo **văn bản được mã hóa H**, cuộn xuống cuối thông báo cảnh báo và nhấp vào nút **Fix** như hình dưới đây:



Bản sửa lỗi cho mỗi cảnh báo văn bản được mã hóa cứng sẽ trích xuất tài nguyên chuỗi cho chuỗi. **Hộp thoại Trích xuất tài nguyên** xuất hiện và bạn có thể nhập tên cho tài nguyên chuỗi. Nhập các tên sau cho tài nguyên chuỗi:

Xâu	Nhập tên sau:
Bánh rán được tráng men và rắc kẹo.	Donuts
Bánh mì kẹp kem có bánh xốp sô cô la và nhân vani.	ice_cream_sandwiches
FroYo là sữa chua đông lạnh tự phục vụ cao cấp.	Froyo

Bố cục bây giờ sẽ trông giống như hình bên dưới.

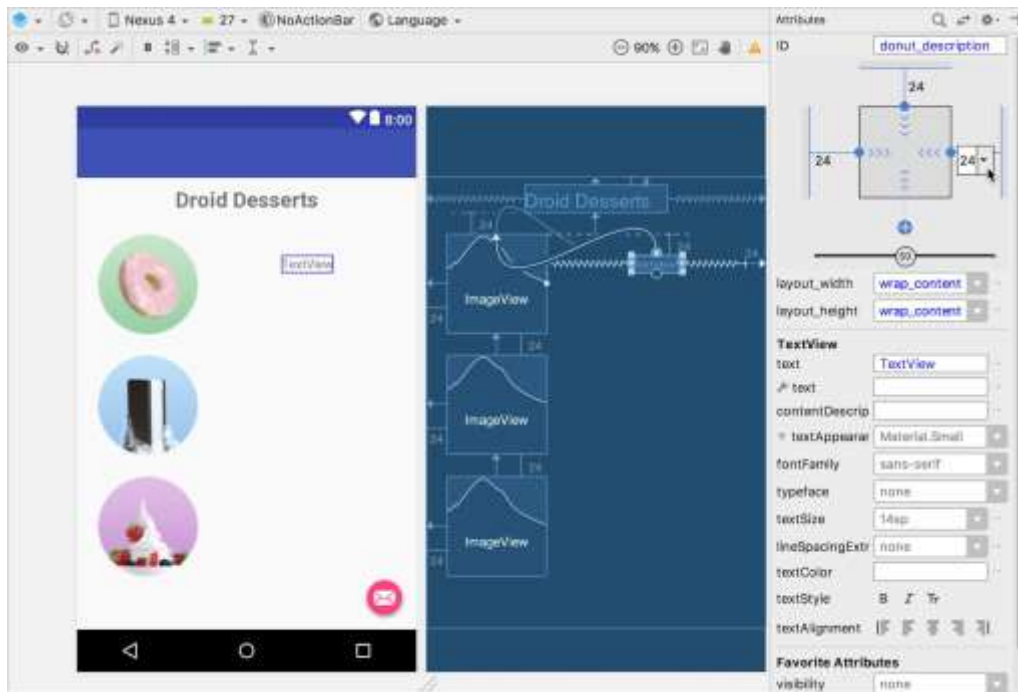


1.3 Thêm mô tả văn bản

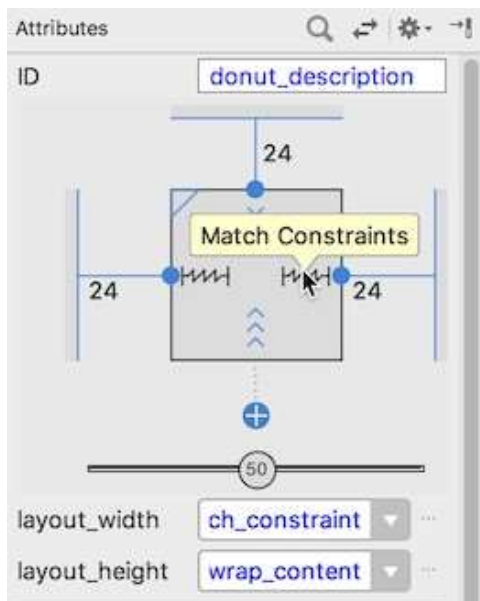
Trong bước này, bạn thêm mô tả văn bản (TextView) cho mỗi món tráng miệng. Vì bạn đã trích xuất tài nguyên chuỗi cho các trường contentDescription cho các phần tử ImageView, bạn có thể sử dụng cùng một tài nguyên chuỗi cho mỗi mô tả TextView.

1. Kéo phần tử TextView vào bố cục.
2. Hạn chế phía bên trái của phần tử ở phía bên phải của donut ImageView và trên cùng của nó ở trên cùng của donut ImageView, cả hai đều có biên độ là **24** (24dp).
3. Hạn chế phía bên phải của phần tử ở phía bên phải của bố cục và sử dụng cùng một lề của

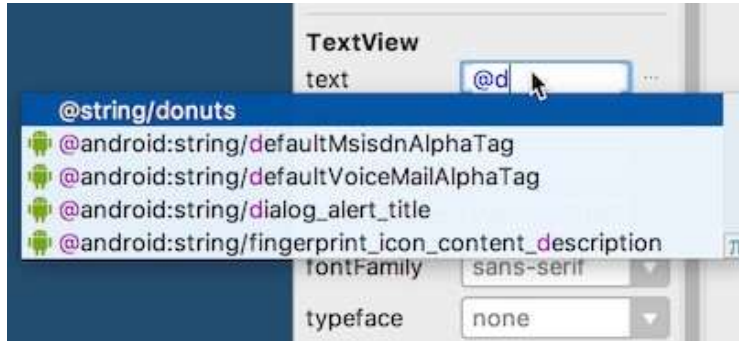
24 (24dp). Nhập **donut_description** cho trường ID trong ngăn **Attributes**. TextView mới sẽ xuất hiện bên cạnh hình ảnh bánh rán như trong hình bên dưới.



4. Trong ngăn **Phân bố A**, thay đổi chiều rộng trong ngăn kiểm tra thành **Match Constraints**:



5. Trong ngăn **Attributes**, bắt đầu nhập tài nguyên chuỗi cho trường text bằng cách đặt đầu nó bằng ký hiệu @: @d. Nhấp vào tên tài nguyên chuỗi (@ string/donuts) xuất hiện dưới dạng gợi ý:



6. Lặp lại các bước trên để thêm một TextView thứ hai bị hạn chế ở phía bên phải và trên cùng của ice_cream ImageView và phía bên phải của nó ở phía bên phải của bố cục. Nhập thông tin sau vào ngăn **Attributes**:

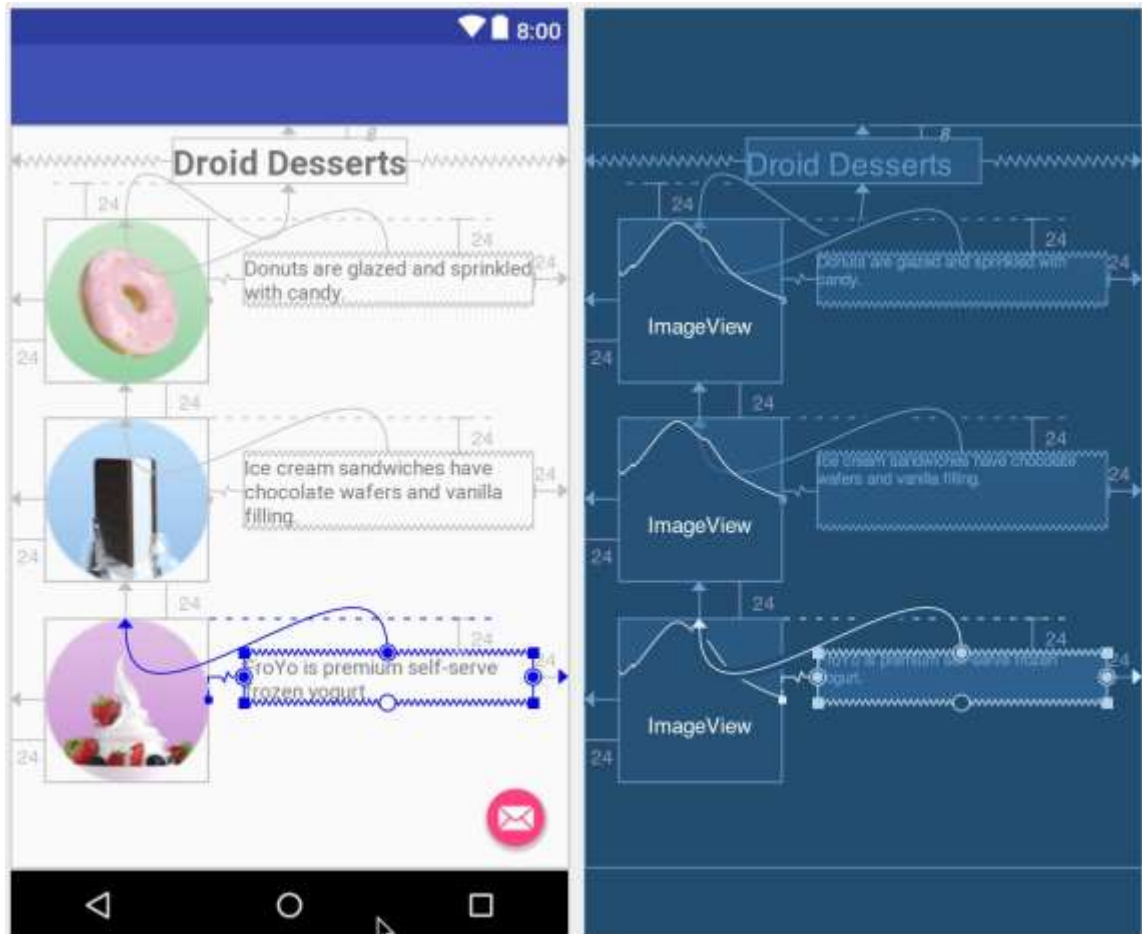
Trường thuộc tính	Nhập như sau:
ID	ice_cream_description
Lề trái, phải và trên	24
layout_width	match_constraint
Nhắn tin	@string/ice_cream_sandwiches

7. Lặp lại các bước ở trên để thêm một TextView thứ ba bị hạn chế ở phía bên phải và trên cùng của froyo ImageView và phía bên phải của nó ở phía bên phải của bố cục. Nhập thông tin sau vào ngăn **Attributes**:

Trường thuộc tính	Nhập như sau:
ID	froyo_description

Lề trái, phải và trên	24
layout_width	match_constraint
Nhấn tin	@string / froyo

Bố cục bây giờ sẽ trông như sau:



Mã giải pháp nhiệm vụ 1

Bố cục XML cho tệp content.xml được hiển thị bên dưới.

```
<?xml version="1.0" encoding="utf-8"?>
```

<android.support.constraint.ConstraintLayout

`xmlns:android="http://schemas.android.com/apk/res/android"`

`xmlns:app="http://schemas.android.com/apk/res-auto"`

```

xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
app:layout_behavior="@string/appbar_scrolling_view_behavior"
tools:context="com.example.android.droidcafe.MainActivity"
    công cụ:showIn="@layout/activity_main">

<Chế độ xem văn bản
    android:id="@+id/textintro"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="@dimen/margin_regular"
    android:text="@string/intro_text"
    android:textSize="@dimen/text_heading"
    android:textStyle="đậm"
    app:layout_constraintLeft_toLeftOf="cha mẹ"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
<Chế độ xem hình ảnh
    android:id="@+id/bánh_rán"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="@dimen/margin_wide"
    android:layout_marginTop="@dimen/margin_wide"
    android:contentDescription="@string/bánh_rán"
    app:layout_constraintStart_toStartOf="cha mẹ"
    app:layout_constraintTop_toBottomOf="@+id/textintro"
    app:srcCompat="@drawable/donut_circle" />

<Chế độ xem hình ảnh
    android:id="@+id/ice_cream"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="@dimen/margin_wide"
    android:layout_marginTop="@dimen/margin_wide"
    android:contentDescription="@string/ice_cream_sandwiches"
    app:layout_constraintStart_toStartOf="cha mẹ"
    app:layout_constraintTop_toBottomOf="@+id/bánh_rán"

```

```

app:srcCompat="@drawable/icecream_circle" />

<Chế độ xem hình ảnh
    android:id="@+id/froyo"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="@dimen/margin_wide"
    android:layout_marginTop="@dimen/margin_wide"
    android:contentDescription="@string/froyo"
    app:layout_constraintStart_toStartOf="cha mẹ"
    ứng dụng:layout_constraintTop_toBottomOf="@+id/ice_cream"
    app:srcCompat="@drawable/froyo_circle" />

<Chế độ xem văn bản
    android:id="@+id/donut_description"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginEnd="@dimen/margin_wide"
    android:layout_marginStart="@dimen/margin_wide"
    android:layout_marginTop="@dimen/margin_wide"
    android:text="@string/bánh_rán"
    app:layout_constraintEnd_toEndOf="cha mẹ"
    app:layout_constraintStart_toEndOf="@+id/donut"
    app:layout_constraintTop_toTopOf="@+id/donut" />
<Chế độ xem văn bản
    android:id="@+id/ice_cream_description"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginEnd="@dimen/margin_wide"
    android:layout_marginStart="@dimen/margin_wide"
    android:layout_marginTop="@dimen/margin_wide"
    android:text="@string/ice_cream_sandwiches"
    app:layout_constraintEnd_toEndOf="cha mẹ"
    app:layout_constraintStart_toEndOf="@+id/ice_cream"
    app:layout_constraintTop_toTopOf="@+id/ice_cream" />
<Chế độ xem văn bản
    android:id="@+id/froyo_description"

```

```
android:layout_width="0dp"
android:layout_height="wrap_content"
android:layout_marginEnd="@dimen/margin_wide"
android:layout_marginStart="@dimen/margin_wide"
android:layout_marginTop="@dimen/margin_wide"
android:text="@string/froyo"
app:layout_constraintEnd_toEndOf="cha mẹ"
app:layout_constraintStart_toEndOf="@+id/froyo"
app:layout_constraintTop_toTopOf="@+id/froyo" />
```

```
</android.support.constraint.ConstraintLayout>
```