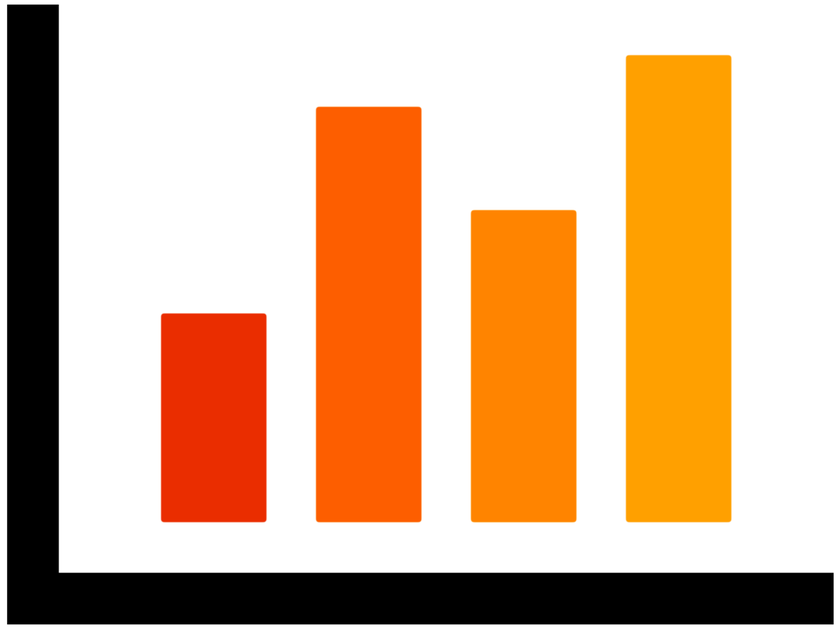


MOBILE APPLICATION DEVELOPMENT

ANDROID (2017)

LECTURE 09: ANIMATION

ANIMATION OVERVIEW

- Animation is often thought of as a way to make visual content dynamic.
- In Android this is often what animation is used for, but Android's animation system is much more flexible and can be used for more than visual updates.
  - Android started out with View animation, which exclusively animates Views.
  - Later Android versions expanded animations to support animating virtually anything, including non-visual data.
  - Android also provides capabilities for animating Drawables using a sequence of still images.

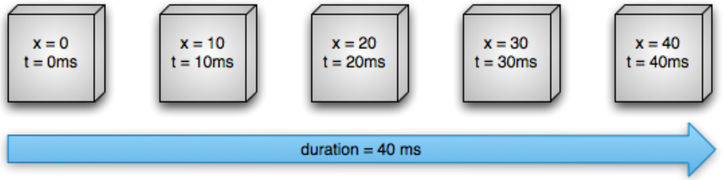
VIEW ANIMATION

- Uses XML or classes like Animation and AnimationSet to specify translations on properties of Views. Intended to present visual animation updates to the user.
- Each Animation / AnimationSet specifies modifications to a variety of View properties that will take place in a given duration.
  - Can transition from one X/Y 'scale' to another.
  - Can change the View's rotation or position.
  - Can change opacity values to make Views appear or disappear slowly.
  - Can use custom interpolators to specify how to link Animation frames together.

DRAWABLE ANIMATION

- Defined in XML or directly in code via the AnimationDrawable class, this kind of animation is intended to cycle a Drawable through many frames, creating an animation.
- This type of animation allows for a lot of flexibility in how a Drawable appears.
  - The programmer may specify completely unrelated images for each frame.
  - Frames may each have their own duration and do not need to animate 'smoothly' if that is not desirable (example: a long hold in the middle of an animation).

VALUE ANIMATION



- Introduced more recently than some animation types, value animation is the most flexible type of animation in Android.
- Animations operate on data rather than graphics, and may have many properties:
  - Animations typically transform a value from a starting minimum to a maximum.
  - Animation 'frames' are (by default) linearly divided across some duration.
  - The programmer may customize the timing of frame updates, and even pause, restart, or reverse animations during the animation.
  - Value animations actually modify the objects they animate to produce animation.

VALUE ANIMATORS

- Value animations are typically accomplished with the ValueAnimator, ObjectAnimator, and AnimatorSet classes.
- These classes use listeners, such as a ValueAnimator.AnimatorUpdateListener, to communicate information to the programmer such as the current value of the animation, the elapsed duration, whether the animation has started, paused, or ended.
  - AnimatorSets allow multiple Animators to be chained together, occur after a delay, or animate simultaneously.
  - Animator listeners may also be used to directly start and stop other Animators.