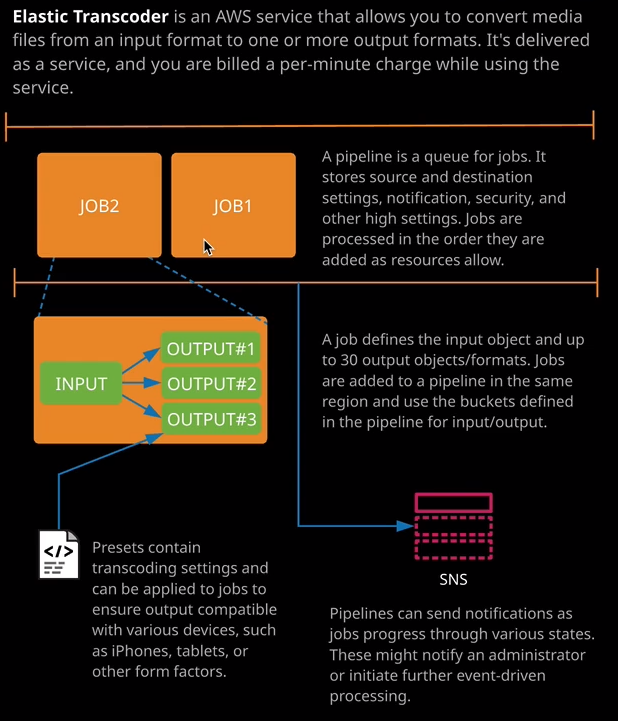
Elastic Transcoder.

**Now Elastic Transcoder, at a very high level, is a service that allows you to convert media files from one format to another**. So you can put your media files on S3 and use Elastic Transcoder to convert from one format to another and it's **delivered as a service** and you only pay for the compute resources that are used during that transcode from the source format to the destination. Now you might use Elastic Transcoder if you've got any video files that you want to deliver to mobile devices or tablets and you're uploading the master copies of those media files. Let's say you're uploading 4k versions of master footage and you want to output a number of smaller media files, maybe files are designed to run on a mobile phone, a tablet, maybe 720p and 1080 p running on laptops or desktops. Well**, Elastic Transcoder is the perfect service to convert between those formats as a service**.



Now within **Elastic Transcoder, the base entity is called a pipeline, and you can think of a pipeline as a queue of jobs. So any jobs that get entered into a pipeline are processed in the order that you add them as compute resources become available**. Okay, so it's the pipeline where you'll be storing a lot of the configuration elements for the transcode process. So you'll need to give the pipeline a name, you'll need to specify an input bucket and it's the **input bucket where any of the source media gets stored**. So if you're uploading 4k master footage, then it's the input bucket where you'll upload that content to, and you'll specify this on the pipeline. You'll also need to specify an **output bucket for any transcoded files and playlists as well as an output bucket for any thumbnails.** Now these could all be the same names, but normally within production pipelines, you're going to have separate buckets for each of these independent assets. You'll be able to specify **notifications for the different events that the pipeline will go through. So when it's processing, when you've got any warnings, any completion events, or any errors.** Once you've got the pipeline created, then it's the pipeline you'll add your jobs to. So if you look at this architecture diagram on the right of my screen, generally, if you got any production level media conversion, you'll always have more than one pipeline. So because the pipeline is a queue any jobs that get entered onto this queue will be processed in order. So most **organizations that I've seen who use Elastic Transcode in production will have pipelines for each different priority of processing needs. So you might have one pipeline for high priority jobs, and the only jobs that are entered into those are the ones that need to be completed as soon as possible. You might have a standard processing queue and then a low processing queue.**

Now **jobs get entered onto those queues and jobs define the input object and up to 30 output objects or formats. So a given job can take one source file, and it can output one or more destination files**. Now the formats for the output components of a job are defined in what's known as **presets** and you can think of presets as configuration files. So we might have a generic preset for 1080p, 720p, 480p. We've got presets for different physical devices. So iPhone, Apple TV, Kindle Fire, Kindle Fire HD, and we've also got traditional TV format so NTSC and PAL, different encoding formats so VP9, HLS. We've got lots of different presets that are predefined and you can also create your own. If you've got specific needs and you can create a preset and use that to control the output format.

Now, with an individual pipeline, **you can pause the processing.** So if you do want to pause any job processing in the pipeline, you've got the ability to do that and as I mentioned earlier in the lesson, when you do create a job, they are processed in the order that they get added to that particular pipeline. So that's critical to understand. **You can define an SNS topic to use for the notifications for that pipeline and then, as jobs move through the pipeline, any notification events will be sent to the SNS topic.**

Now for the exam, just awareness of this product is enough. I don't expect that at the associate level that you'll get any detailed questions on Elastic Transcoder. You might face questions, for example, where you need to transcode media files from one format to the other. You might be presented with a number of options maybe EC2, maybe Lambda, maybe Elastic Transcoder, and you'll have to select the best product based on the criteria. So the question might present a scenario **where you need to keep costs down, where you need to have low admin overhead, where you don't want any base compute costs, and in that case, Elastic Transcoder is a perfect option. Lambda may not have the runtime required to convert large files and EC2 will have a base processing cost.** So, Elastic Transcoder, in most cases, is going to be the correct answer for any transcoding of media files within AWS.

Now, whilst I've been giving you a tour of the console, this has been a manual process, and the jobs that you create inside the Elastic Transcoder console are going to be manual but you can automate this process so you could create S3 events that respond to media objects getting uploaded to a bucket. That S3 event could cause a Lambda function to be invoked and that Lambda function could create an Elastic Transcode jobs. You can create an event driven pipeline that automatically converts media files from one format to another the minute they're uploaded to an S3 bucket and that could be a really powerful solution **for a serverless media transcoding pipeline**. So it'll upload the objects to S3, S3 event triggers invokes a Lambda function, the Lambda function creates the transcoding job, and that handles the conversion from the source to the destination format.