**Particiones de equivalencia**

Una forma bien simple de comprender qué es una partición de equivalencia es pensar en la siguiente analogía:

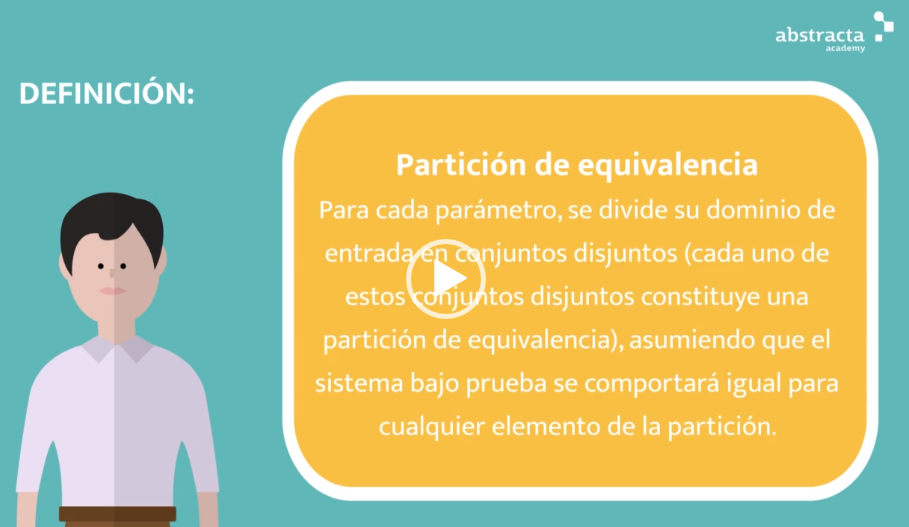
Tenemos 3 bolsas de caramelos, cada una con un sabor diferente, pero las bolsas a simple vista son todas iguales. Entonces no sabemos que hay en cada bolsa. Cómo podemos hacer para descubrir qué sabor hay en cada bolsa? Muy simple: tomamos un caramelo de cada bolsa y lo probamos. Con probar uno ya alcanza, no necesitamos probar todos los caramelos de la bolsa, pues si el que sacamos es de naranja, todos los demás también serán de naranja. Y entonces podemos deducir que todos los caramelos son equivalentes. No importa cual tomemos, todos tendrán el mismo sabor.



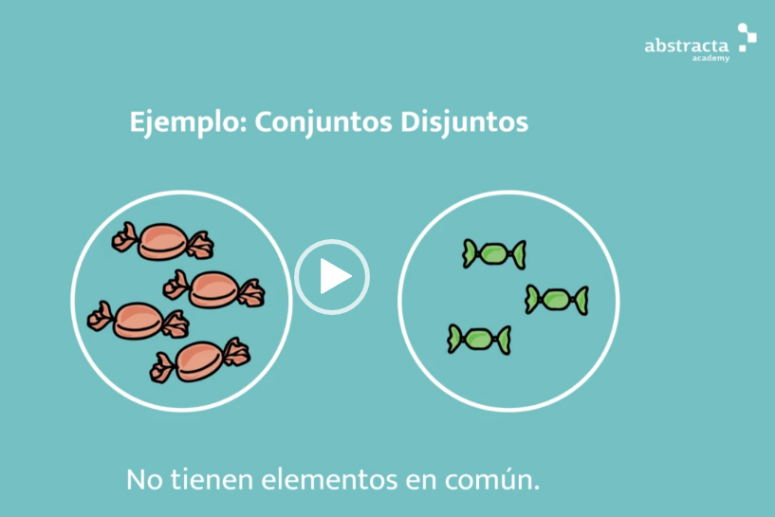
Ahora llevemos esto a nuestro objetivo principal de probar: considerar la menor cantidad de pruebas que me garanticen la mayor calidad del sistema. Es imposible probar todas las posibles situaciones de datos y variables que pueden tener un sistema. Entonces vamos a tener que identificar cuáles son los elementos, los valores que pertenecen a la misma bolsa y sólo tomar un elemento de esa bolsa. No vamos a probar todos los elementos que tengan un comportamiento similar, sino que tomaremos uno de muestra al que llamaremos **representante** y sólo probaremos con ese, ya que el resto se comporta de forma equivalente. De ahí es que viene el nombre de particiones de equivalencia.



**Definición de particiones de equivalencia:** la técnica dice que para cada parámetro se divide su dominio de entrada en conjuntos disjuntos (cada uno de estos conjuntos constituye una partición de equivalencia), y se asume que el sistema bajo pruebas se comporta igual para cualquier elemento de esa partición de ese conjunto.



Dos conjuntos son disjuntos cuando no tienen elementos en común.



Y si volvemos al ejemplo de los caramelos no hay una bolsa que tenga algunos caramelos de naranja y otros caramelos de limón, sino que partimos de la premisa que todas las bolsas tienen caramelos del mismo sabor.

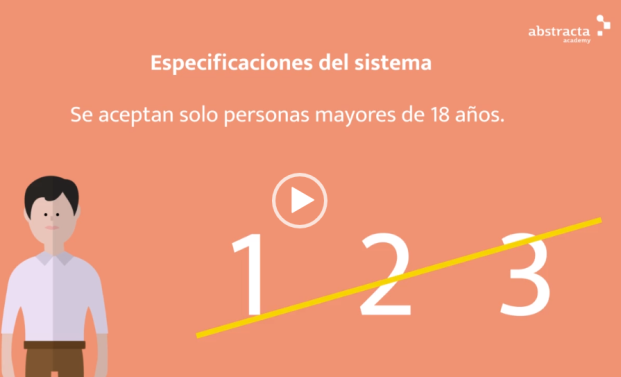


Esto significa que tenemos que ser cuidadosos al momento de armar los conjuntos: que sean disjuntos entre sí. Y tenemos que asegurarnos que todos los valores que ponemos en la misma bolsa son equivalentes.

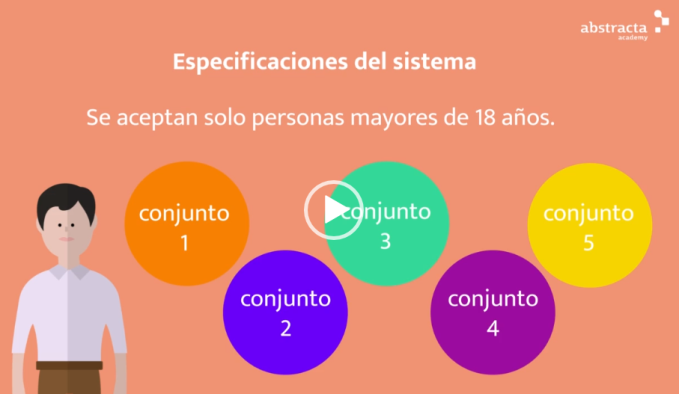


**Ejemplo de particiones de equivalencia**

**Ejemplo aplicado al testing:** imaginemos que nos entregan una especificación de un sistema que sirve para realizar diversos trámites, donde se indica que se aceptan sólo personas mayores de 18 años. Pensemos con qué valores probaremos esta aplicación teniendo en cuenta lo que vimos hasta ahora. No tiene sentido probar la aplicación con una persona de 1 año, 2 años, 3 años y así consecutivamente.



Lo que debemos hacer es identificar cuáles son las particiones de equivalencia, los distintos conjuntos de valores que tendrán comportamiento equivalente, o sea de esa forma distinguir cuales son las bolsas de caramelos. Entonces qué particiones deberíamos considerar para la edad de la persona en éste sistema de trámites?



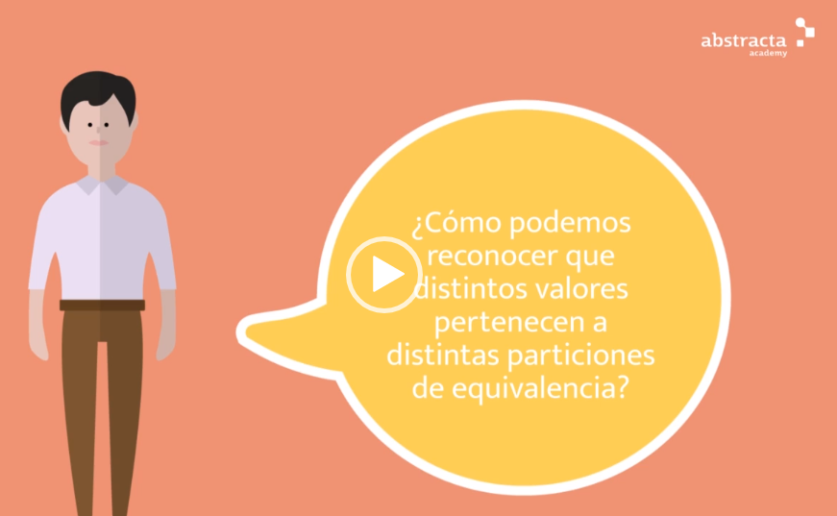
Haz una pausa en el video, lo piensas un momento y lo escribas. La solución es muy simple: tenemos que distinguir las personas mayores de 18 y las personas que tienen entre 0 y 18 años. Se te ocurre alguna otra? Piensa en situaciones menos comunes. Acá una forma de analizarlo es quizás con la siguiente pregunta: qué sucede con los menores de 0 años? O qué pasa con los mayores de 150 por ejemplo? Tiene sentido hablar de personas mayores de esa edad? Quizás se te ocurra alguna más… y si en lugar de probar con números la persona se equivoca y pone una letra?



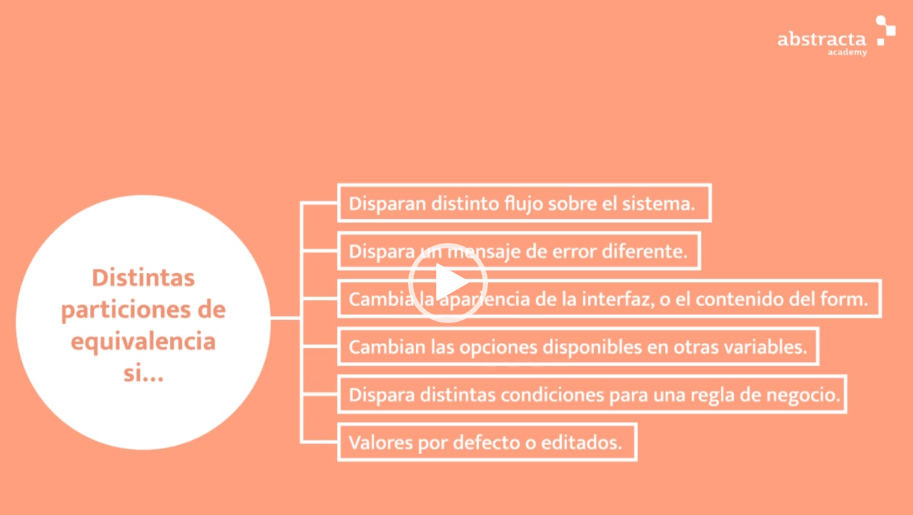
Tal vez estarás pensando que es imposible que una persona de más de 150 años intente hacer un trámite en este sistema, o es imposible que un usuario ponga una letra en lugar de su edad, pero acá lo que estaríamos probando es qué pasa si el usuario se equivoca. Por ejemplo, si está accediendo a la aplicación desde su móvil y está ingresando su edad con unos botones que son muy pequeños y por eso presiona algo mal o quizás interpretó que tenía que escribir cincuenta y ocho en lugar del número 58 o se le escapó un número de más y en lugar de poner 25 puso 255. Nosotros al probar éstas situaciones estaremos verificando que el **sistema es robusto**, que puede defenderse de errores de usuarios, y no es que los usuarios sean malos y que vayan a poner estos valores incorrectos a propósito, sino que los usuarios en diferentes contextos pueden llegar a equivocarse y nosotros tenemos que estar preparados para eso.



Bueno, luego de esto quizá te estés preguntando cómo podemos reconocer que distintos valores pertenecen a distintas particiones de equivalencia? A distintas bolsas de caramelos? Veamos algunas cosas que nos dan pistas para responder a ésta pregunta.



Por ejemplo podríamos pensar que si distintos valores nos disparan distinto flujo sobre el sistema entonces pertenecerán a distintas particiones de equivalencia, o si disparan un mensaje de error diferente, o por ejemplo si cambia la apariencia de la interfaz o el contenido de un formulario o cambian las opciones disponibles en otras variables, o por ejemplo si dispara distintas condiciones para una regla de negocio o también en ocasiones si utilizamos los valores por defecto o si modificamos esos valores por defecto.



A continuación pondremos en práctica esta técnica con un ejercicio, para luego poder dedicarnos a estudiar otras técnicas más avanzadas que nos permiten probar cualquier sistema con mayor profundidad.