

# Piecewise Functions

- Sometimes we want to build functions that act differently for different inputs. For example, suppose a business charges \$10/pizza, but only \$5 for orders of six or more. How could we write a function that computes the total price based on the number of pizzas?
- In math, **Piecewise Functions** are functions that can behave one way for part of their Domain, and another way for a different part. In our pizza example, our function would act like  $cost(pizzas) = 10 * pizzas$  for anywhere from 1-5 pizzas. But after 5, it acts like  $cost(pizzas) = 5 * pizzas$ .
- Piecewise functions are divided into "pieces". Each piece is divided into two parts:
  1. How the function should behave
  2. The domain where it behaves that way
- Our programming language can be used to write piecewise functions, too! Just as in math, each piece has two parts:

```
fun cost(pizzas):  
  ask:  
    | pizzas < 6 then: 10 * pizzas  
    | pizzas >= 6 then: 5 * pizzas  
  end  
end
```

- Piecewise functions are powerful, and let us solve more complex problems. We can use piecewise functions in a video game to add or subtract from a character's x-coordinate, moving it left or right depending on which key was pressed.