

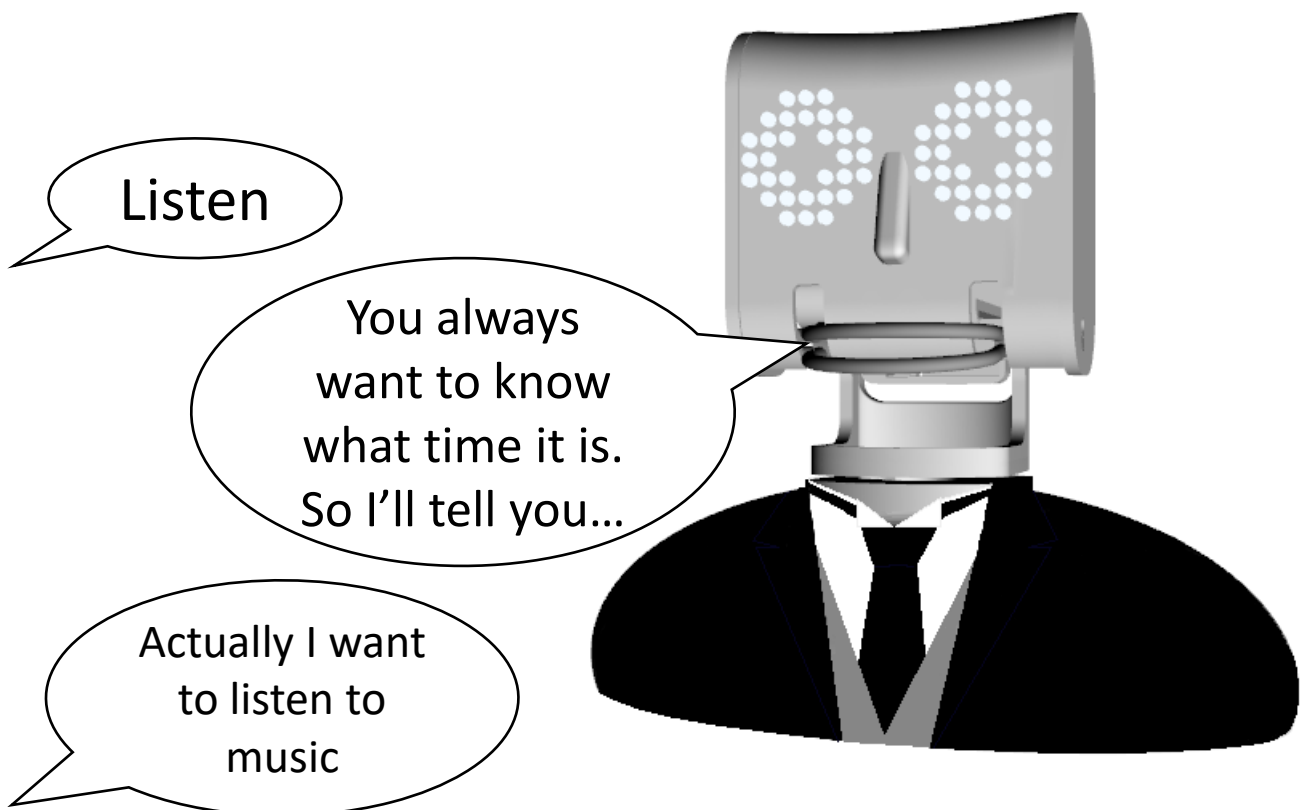
# personal Assistant part 3

Program your  
robot to be a  
smart assistant

In the first two parts of this resource we added the various skills we want our virtual assistant to have.

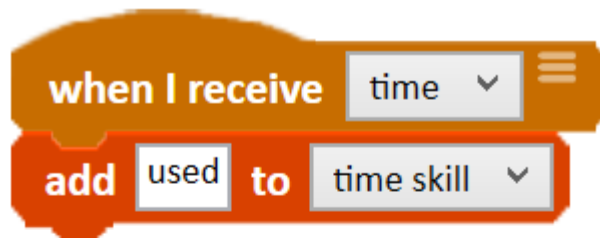
All we have to do is ask and the robot will run the appropriate skill. But what if we didn't have to ask?

What if our virtual assistant learned which skill we asked for most often and ran it autonomously? Can you foresee any potential problems that functionality?



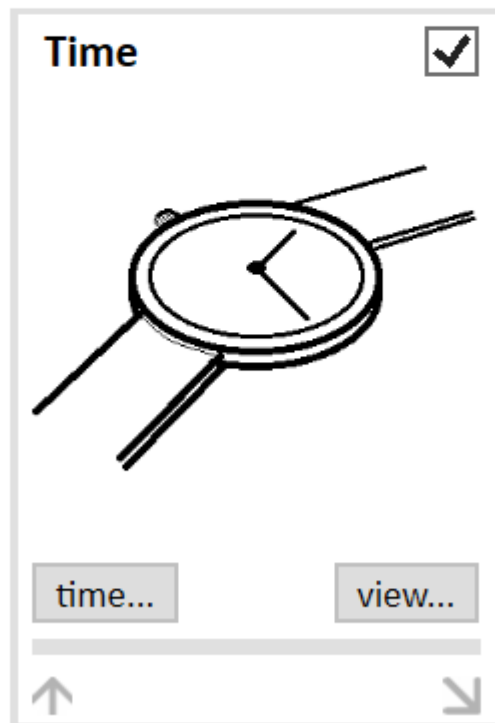
So far the way our virtual assistant has worked is, every time you say the name of a skill, that word is broadcast to a multiblock and the skill is triggered.

To teach our robot which skills we are using most frequently we are going to add another **when I receive** block and create a separate list for each skill. Every time a skill is used it will add a count to its list.



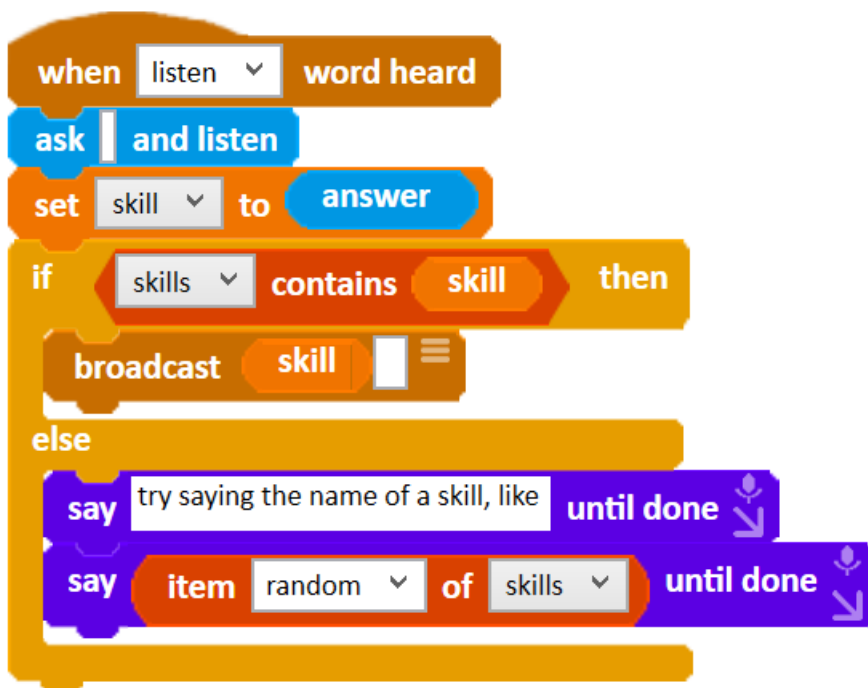
When "time" is received the word "used" will be added to the time skill list

time skill: used



Now we have a way for the robot to count how many times we use each skill. But how can we use that knowledge to make the robot run the most popular skill without being asked to?

Here is the existing code, how would you alter it?



```
when listen word heard
ask and listen
set skill to answer
if skills contains skill then
  broadcast skill
else
  say try saying the name of a skill, like until done
  say item random of skills until done
```

The image shows a Scratch script with the following blocks:

- when listen word heard** (orange)
- ask and listen** (blue)
- set skill to answer** (orange)
- if skills contains skill then** (yellow)
- broadcast skill** (orange)
- else** (yellow)
- say try saying the name of a skill, like until done** (purple)
- say item random of skills until done** (purple)

We've positioned an **if \_ then** block in the code ahead of the **ask \_ and listen** block. So when the conditions of the **if \_ then** block are met, it will run first.

When the length of the time skill list is greater than two, the robot will automatically trigger the time skill with a **broadcast and wait** block

```
when listen word heard
  if length of time skill > 2 then
    say "You have used the time skill a few times already. I'll do it again for you" until done
    broadcast time and wait
    delete all of time skill
  ask and listen
  set skill to answer
  if skills contains skill then
    broadcast skill
  else
    say "try saying the name of a skill, like" until done
    say "item random of skills" until done
```

After the predicted skill has run, the rest of the program will run allowing you to choose a skill

Can you see any problems with a virtual assistant that behaves like this?

Are these the same problems that you predicted at the start?

Can you think of any solutions to these problems? Are there any advantages to your robot knowing what skills you use most frequently?

Try adding skill counters for the other skills that you have programmed for your robot.

