Name_____

Lesson 5: Worksheet 5.1 – Play tones

In this activity, you need to write a program to make Edison play a musical note and learn how Edison plays sounds in a program.

You can play individual musical notes through Edison's small speaker using the Ed.PlayTone() function in EdPy.

The Ed.PlayTone() function takes two input parameters: the note and the duration. The note determines what note to play and the duration determines the given length of time the note should be played.

This list includes the possible parameter values:

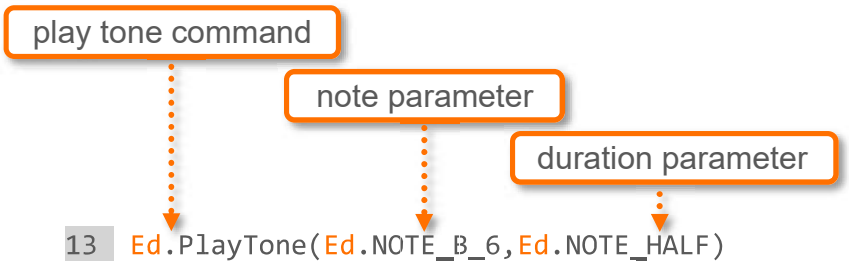
note

Parameter input options	Plays musical note
Ed.NOTE_A_6	low A
Ed.NOTE_A_SHARP_6	low A sharp
Ed.NOTE_B_6	low B
Ed.NOTE_C_7	C
Ed.NOTE_C_SHARP_7	C sharp
Ed.NOTE_D_7	D
Ed.NOTE_D_SHARP_7	D sharp
Ed.NOTE_E_7	E
Ed.NOTE_F_7	F
Ed.NOTE_F_SHARP_7	F sharp
Ed.NOTE_G_7	G
Ed.NOTE_G_SHARP_7	G sharp
Ed.NOTE_A_7	A
Ed.NOTE_A_SHARP_7	A sharp
Ed.NOTE_B_7	B
Ed.NOTE_C_8	high C
Ed.NOTE_REST	rest

duration

Parameter input options	Plays note for
Ed.NOTE_SIXTEENTH	125 milliseconds
Ed.NOTE_EIGHTH	250 milliseconds
Ed.NOTE_QUARTER	500 milliseconds
Ed.NOTE_HALF	1,000 milliseconds
Ed.NOTE_WHOLE	2,000 milliseconds

Let's take a closer look at the play tone function in a program:



Using the parameter values tables as a reference, can you work out what the program will do?

This program will play a low B note for a duration of 1 second.

Name_____

Your turn:

Task 1: Play a note

Write the following program:

```
1
2  #-------------Setup----------------
3
4  import Ed
5
6  Ed.EdisonVersion = Ed.V2
7
8  Ed.DistanceUnits = Ed.CM
9  Ed.Tempo = Ed.TEMPO_MEDIUM
10
11 #--------Your code below-----------
12
13 Ed.PlayTone(Ed.NOTE_A_SHARP_7,Ed.NOTE_HALF)
14
```

Download and test the program to see what it sounds like.

Task 2: Play a note, then drive? Or play a note while driving?

When Edison plays sounds, it does this in the background. This means that as soon as Edison starts playing the sound, the program will move onto the next line of code. The sound will keep playing 'in the background' while Edison continues on with the program.

If you want Edison to wait for the sound to finish, you need to use the Ed.ReadMusicEnd() function in a 'while' loop.

Write the following program:

```
1
2  #-------------Setup----------------
3
4  import Ed
5
6  Ed.EdisonVersion = Ed.V2
7
8  Ed.DistanceUnits = Ed.CM
9  Ed.Tempo = Ed.TEMPO_MEDIUM
10
11 #--------Your code below-----------
12 Ed.PlayTone(Ed.NOTE_C_8, Ed.NOTE_WHOLE)
13 while Ed.ReadMusicEnd()==Ed.MUSIC_NOT_FINISHED:
14     pass
15 Ed.Drive(Ed.FORWARD, Ed.SPEED_6, 5)
16
```

Download and test the program.

Name_____

1. Describe what happened when you ran this program.

_____

_____

_____

2. Look at line 13 and 14 of the program. Remember that expressions compare the left side to the right side of the notation in the expression. What is this loop doing?

_____

_____

_____

Write the following program:

```
1
2  #-------------Setup----------------
3
4  import Ed
5
6  Ed.EdisonVersion = Ed.V2
7
8  Ed.DistanceUnits = Ed.CM
9  Ed.Tempo = Ed.TEMPO_MEDIUM
10
11 #--------Your code below-----------
12 Ed.PlayTone(Ed.NOTE_C_8, Ed.NOTE_WHOLE)
13 Ed.Drive(Ed.FORWARD, Ed.SPEED_6, 5)
14
```

Download and test the program.

3. Describe what happened when you ran this program.

_____

_____

_____

4. Why did this program behave differently than the last program?

_____

_____

_____

Name_____

Lesson 5: Worksheet 5.2 – Make an alarm

In this activity, you need to write a program to make Edison play an alarm at a frequency you specify.

Using the Ed.PlayTone() function, you can customise the exact frequency of the sound that Edison's speaker produces by using numbers and variables.

Frequency in acoustics

As you may know, sound travels in waves called sound waves. Acoustics, the branch of physics that deals with sound and sound waves, looks at everything to do with sound, including how to measure it.

One way to measure sound is by measuring frequency. Frequency is the number of waves passing a point in a certain period of time.

Frequency is most often measured in cycles per second (cycle/sec). The base unit for frequency is hertz, abbreviated Hz.

One hertz is equal to one complete wave per second.

Did you know? The human hearing range is 20 Hz ~ 20000 Hz.

Frequency and period

In addition to the musical notes that are pre-set in EdPy, we can also program Edison to play sounds with different frequencies.

To do this, we convert frequencies into periods, which Edison can understand.

A period is how long it takes an acoustic wave to complete a full cycle. Since we are using hertz, we measure frequency in cycles per second.

In acoustics, when period increases, frequency decreases.

Let's look at some examples of how frequency and period relate:

- If a wave has a period of 0.5 seconds, it has a frequency of 2Hz because it can complete 2 cycles in 1 second.
- If a wave has a period of 2 seconds, it has a frequency of 0.5Hz because it can only complete half of a cycle in 1 second.

Converting frequency to period for your program

To get Edison to play a custom frequency, we need to work out the value of the period. This is the number we input into the 'note' parameter in Ed.PlayTone().

Name_____

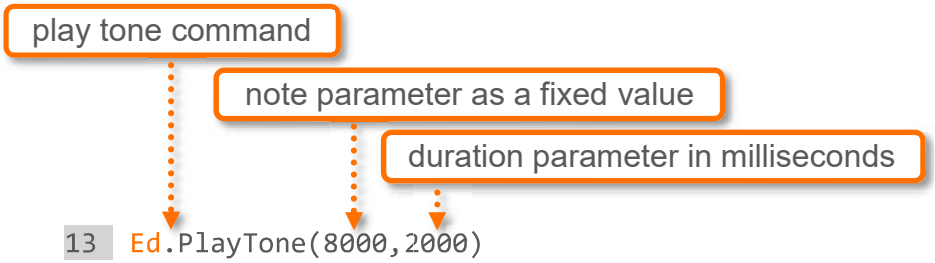
To convert a frequency into a period, divide the number 8,000,000 by the desired frequency. For example, to play a 1kHz (1000 cycles per second) sound:

$$\frac{8000000}{1000} = 8000$$

Your turn:

Task 1: Play a custom tone

Write the following program:



Download and test it to hear what this program sounds like.

Task 2: Play an alarm

In this program, we want Edison to play notes of increasing period.

To make the alarm program, you will need to use a 'for' loop, variables and the range() function. You also need to nest a 'while' loop into the program.

Write the following program:

```
1
2  #-------------Setup----------------
3
4  import Ed
5
6  Ed.EdisonVersion = Ed.V2
7
8  Ed.DistanceUnits = Ed.CM
9  Ed.Tempo = Ed.TEMPO_MEDIUM
10
11 #--------Your code below-----------
12
13 for i in range(33):
14     Ed.PlayTone(100+(i*100), 1000)
15     while Ed.ReadMusicEnd()==Ed.MUSIC_NOT_FINISHED:
16         pass
17
```

Download it and test what this program sounds like.

Name_____

1. What do you hear from the robot? Why is this happening?

_____

_____

_____

An important skill in programming is being able to 'trace' through a program to understand what is happening. Programmers perform a code trace as a method for hand simulating the execution of their code to verify that it works correctly before compiling it manually.

Tracing involves stepping through the program line by line, recording important values. It is often done to help find errors or 'bugs' in code, but it is also useful when you just need to understand what is happening in a program.

Try to 'trace' through what is happening in the program and answer the following questions about the program.

2. Fill in the following table by calculating the period parameter for each given value of 'i' in the above code. The first value is filled in for you.

Value of i	Period parameter [the 1st input parameter to PlayTone()]
0	100
1	
2	

3. What is the maximum value of i?

   _____

4. What is the maximum value of the period parameter input to the PlayTone() function?

   _____

5. How many tones are played?

   _____

Name_____

Try it!

The application of acoustics in technology is called acoustical engineering.

Try some acoustical engineering of your own. Experiment with modifying the parameters to PlayTone() to make the program play a different combination of sounds.

Name_____

Lesson 5: Worksheet 5.3 – Play a tune

In this activity, you need to write a program to make Edison play a musical tune.

You can get Edison to play a tune using the Ed.PlayTune() function and a special type of input called a 'string.'

Using a string to play a tune

In Python, a 'string' is a list of characters in order. A 'character' is anything you can type on the keyboard like a letter, a number, or a special character like $ or #. For example, 'Meet Edison' is a string, 11 characters long (10 letters and 1 space).

In the EdPy app, we need to use a string to play a musical tune. We call this a 'tune string.'

Tune strings are a special string of characters that represent particular tunes. Tune strings are made up of notes and duration inputs, which are represented by single characters.

A tune string looks like this: "ndndndndnd…ndz" where n is a note from the notes table and d is duration from the duration table:

Notes Table

String character	Plays musical note
m	low A
M	low A sharp
n	low B
c	C
C	C sharp
d	D
D	D sharp
e	E
f	F
F	F sharp
g	G
G	G sharp
a	A
A	A sharp
b	B
o	high C
R	rest
z	end of tune

Duration Table

String character	Plays
1	whole note
2	half note
4	quarter note
8	eighth note
6	sixteenth note

All tune strings must end with the 'z' character to end correctly.

To create a tune string, you need to call the function Ed.TuneString(), which has two input parameters. The size of the string (in other words, the number of characters in the string) is the first parameter, and the actual string you want to play is the second parameter.

You can change the speed your tune plays by changing the Ed.Tempo variable in the Setup code.

Name_____

Your turn:

Write the following code to play the tune 'Mary Had a Little Lamb':

```
1
2  #-------------Setup----------------
3
4  import Ed
5
6  Ed.EdisonVersion = Ed.V2
7
8  Ed.DistanceUnits = Ed.CM
9  Ed.Tempo = Ed.TEMPO_MEDIUM
10
11 #--------Your code below-----------
12 maryLamb = Ed.TuneString(53,"e4d4c4d4e4e4e2d4d4d2e4g4g4e4d4c4d4e4e4e4e4d4d4e4d4c1z")
13
14 Ed.PlayTune(maryLamb)
15 while Ed.ReadMusicEnd() == Ed.MUSIC_NOT_FINISHED:
16     pass
17
```

This is the tune string in the program:

"e4d4c4d4e4e4e2d4d4d2e4g4g4e4d4c4d4e4e4e4e4d4d4e4d4c1z"

Experiment with changing the Ed.Tempo value in the Setup code.

1. What are the different values that Ed.Tempo can take?

Hint: Remember you can use the autocomplete feature in the EdPy. Try typing 'Ed.TEMPO' and see all the possible values for Ed.TEMPO the autocomplete brings up.

_____

_____

2. Which Ed.TEMPO value will make the tune play the fastest?

_____

3. Modify your program to only play some part of the tune. Describe the changes you had to make to your program to only play a part of the tune.

_____

_____

_____

_____

_____

Name_____

Lesson 5: Worksheet 5.4 – Make your robot dance

In this activity, you will write a program to make your robot dance.

In most good dance performances, there are some moves or actions which are repeated. You can make your Edison repeat actions in a dance routine by using the 'for' loop.

The 'shimmy' is a dance move where you hold your body still and quickly move your shoulders back and forth.

Look at the following program which will make your Edison robot do a version of a shimmy:

```
1
2  #-------------Setup----------------
3
4  import Ed
5
6  Ed.EdisonVersion = Ed.V2
7
8  Ed.DistanceUnits = Ed.CM
9  Ed.Tempo = Ed.TEMPO_MEDIUM
10
11 #--------Your code below-----------
12
13 #Set up some variables
14 turnSpeed = Ed.SPEED_9
15 degreesToTurn = 20
16 numberOfTwists = 3
17
18 #Now shimmy!
19 Ed.Drive(Ed.SPIN_RIGHT,turnSpeed,degreesToTurn/2)
20 for i in range(numberOfTwists):
21     Ed.Drive(Ed.SPIN_LEFT,turnSpeed,degreesToTurn)
22     Ed.Drive(Ed.SPIN_RIGHT,turnSpeed,degreesToTurn)
23 Ed.Drive(Ed.SPIN_LEFT,turnSpeed,degreesToTurn/2)
24
```

This program uses variables so that it will be easy to change the turning speed, the number of twists in the dance and the degrees Edison will turn.

Both lines 13 and 18 start with '#' which means these lines are comment code lines added to make it easier for us to read the program. Remember, Edison will skip any line that starts with '#'.

Look at lines 19 and 23. In these lines, we are doing a mathematical calculation in our code to make Edison turn only half the number of degrees.

Your turn:

Write the program.

Download the program to your Edison and run it to see the dance in action.

Name_____

1. How many times does the robot turn to the left?

   _____

2. How many times does the robot turn to the right?

   _____

3. The first turn to the right is only half the distance of all the turns inside the 'for' loop because this line has the input parameter 'degreesToTurn/2'. Why do want this line in the program? Try removing the maths (the /2) and run the program again. What do you notice? (*Hint:* look at how far Edison moves left compared to the start point.)

_____

_____

_____

_____

Try it!

Experiment with the program. Try changing the variables to change the way Edison dances. Change the number of degrees Edison will turn, the speed Edison will turn, the number of twists in the dance or all three!

Name_____

Lesson 5: Worksheet 5.5 – Challenge! Dance to music

Dancing is more fun with music! In this activity, you will write a program combining dance moves with some tones or a tune.

Your turn:

Write and run the following program that combines a 'shimmy' dance with some tones:

```
#-------------Setup----------------

import Ed

Ed.EdisonVersion = Ed.V2

Ed.DistanceUnits = Ed.CM
Ed.Tempo = Ed.TEMPO_MEDIUM

#--------Your code below-----------

#Set up my variables
turnSpeed = Ed.SPEED_9
degreesToTurn = 60
numberOfTwists = 3

#Now dance to the music!
Ed.Drive(Ed.SPIN_RIGHT, turnSpeed, degreesToTurn/2)
Ed.PlayTone(Ed.NOTE_C_7, Ed.NOTE_SIXTEENTH)
for i in range(numberOfTwists):
    Ed.Drive(Ed.SPIN_LEFT, turnSpeed, degreesToTurn)
    Ed.PlayTone(Ed.NOTE_A_7, Ed.NOTE_SIXTEENTH)
    Ed.Drive(Ed.SPIN_RIGHT, turnSpeed, degreesToTurn)
    Ed.PlayTone(Ed.NOTE_C_7, Ed.NOTE_SIXTEENTH)
Ed.Drive(Ed.SPIN_LEFT, turnSpeed, degreesToTurn/2)
Ed.PlayTone(Ed.NOTE_A_7, Ed.NOTE_SIXTEENTH)

```

Now design your own dance for your Edison, adding some tones or using a tune string. Can you synchronise it so Edison dances in time with the music?

1. Describe your robot's dance moves. Is there anything in your program you really liked? If so, describe it.

_____

_____

_____

_____

_____

Name_____

2. What combination of tones or notes did you play along with your dance?

_____

_____

_____

_____