

## Jingle Bells (simple version) με χρήση python - Θετική ομάδα - Δεκέμβριος 2021

Για να μπορέσετε να γράψετε τον κώδικα για αναπαραγωγή της μελωδίας είναι απαραίτητη η εγκατάσταση περιβάλλοντος ανάπτυξης προγράμματος. Εδώ χρησιμοποιείται το Spyder.

Είναι διαθέσιμο μέσα από το περιβάλλον του συστήματος **Anaconda** για Windows.

<https://www.anaconda.com/products/individual>

Αφού ολοκληρώσετε την εγκατάσταση του προγράμματος στο μενού έναρξης μπορείτε να κάνετε απευθείας εκκίνηση του περιβάλλοντος **Spyder**.

```
jinkle_bells.py x
1  # -*- coding: utf-8 -*-
2  """
3  Jingle Bells (easy version) σε μία οκτάβα
4
5  @author: θετική ομάδα "Πολύτροπη Αρμονία"
6  """
7
8  a={'Do':262,'Re':294,'Mi':330,'Fa':349,'Sol':392,'La':440,'Si':494,'P':28000}
9  music=[['Mi',4],['Mi',4],['Mi',2],['Mi',4],['Mi',4],['Mi',2],
10         ['Mi',4],['Sol',4],['Do',4],['Re',4],['Mi',1],
11         ['Fa',4],['Fa',4],['Fa',4],['Fa',4],['Fa',4],['Mi',4],['Mi',4],['Mi',16],['Mi',16],
12         ['Mi',4],['Re',4],['Re',4],['Mi',4],['Re',4],['P',4],['Sol',4],['P',4],
13         ['Mi',4],['Mi',4],['Mi',2],['Mi',4],['Mi',4],['Mi',2],
14         ['Mi',4],['Sol',4],['Do',4],['Re',4],['Mi',1],
15         ['Fa',4],['Fa',4],['Fa',4],['Fa',4],['Fa',4],['Mi',4],['Mi',4],['Mi',16],['Mi',16],
16         ['Sol',4],['Sol',4],['Fa',4],['Re',4],['Do',1]]
17
18  from winsound import Beep
19  def play_musiclist(music):
20      for x in music:
21          [note,time]=x
22          freq=a[note]
23          duration=int(1920/time)
24          print("Note=",note,"(",freq,"Hz)",end="")
25          print("Dur=1/",time,"(",duration,"msec)")
26          Beep(freq,duration)
27  play_musiclist(music)
```

### κώδικας

**Γραμμή 1:** Εισάγουμε με άγκιστρα το λεξικό **a**, το οποίο αποτελείται από 8 στοιχεία. Κάθε στοιχείο του λεξικού είναι ζεύγος «κλειδί : τιμή», δηλαδή στην περίπτωσή μας «νότα της κλίμακας : συχνότητα της νότας σε Hz». Το τελευταίο στοιχείο αντιστοιχεί σε «παύση P : συχνότητα 28000Hz», η οποία ως υπέρηχος δεν ακούγεται.

**Γραμμή 9 ως και 16:** εισάγουμε με αγκύλες τη λίστα **music**. Κάθε στοιχείο **x** της λίστας είναι μια άλλη λίστα που περιέχει δύο στοιχεία. Το πρώτο είναι η νότα **[note]**, το δεύτερο η χρονική αξία της νότας **[time]** εκφρασμένη με τον παρονομαστή του κλάσματος ( 1 για ολόκληρο, 2 για ½(μισό), 4 για ¼, 16 για 1/16).

**Γραμμή 18:** Από τη βιβλιοθήκη **winsound** της python εισάγουμε το στοιχείο **Beep** το οποίο αντιστοιχεί στο χαρακτηριστικό «μπιπ» με το οποίο θα αναπαράγεται η μελωδία στον υπολογιστή.

**Γραμμή 19:** Με τη χρήση της συνάρτησης **play\_musiclist()** μπορούμε να μετατρέψουμε σε μουσική τη λίστα **music**.

**Γραμμή 20:** με τη συγκεκριμένη εντολή εξασφαλίζουμε ότι η αναπαραγωγή της μουσικής θα γίνει για κάθε στοιχείο **x** της λίστας **music**.

**Γραμμή 21:** ορίζουμε ότι κάθε στοιχείο της λίστας **music** που το ονομάσαμε **x** είναι μια άλλη λίστα με δύο στοιχεία [**note,time**], όπως ειπώθηκε.

**Γραμμή 22:** Κατασκευάζουμε μια νέα λίστα **freq** στην οποία κάθε στοιχείο **note** της υπολίστας στη λίστα **music** θα αντικατασταθεί από την αντίστοιχη συχνότητα στο λεξικό **a**.

**Γραμμή 22:** Εισάγουμε τη χρονική διάρκεια **duration** της αναπαραγωγής κάθε νότας με «μπιπ» να είναι ίση με το ακέραιο (int) του πηλίκου  $1920/\text{time}$ . Τα 1920msec τα ορίσαμε αυθαίρετα στο κομμάτι να είναι η χρονική διάρκεια του ολόκληρου, ώστε να διαιρείται ακριβώς με τις υπόλοιπες χρονικές αξίες ( για το μισό  $1920/2=960\text{msec}$ , για το τέταρτο  $1920/4=480\text{msec}$ , για το όγδοο  $1920/8=240\text{msec}$ , για το δέκατο έκτο  $1920/16=120\text{msec}$ ....)

**Γραμμή 24 και 25:** Με την εντολή **print** κατά την αναπαραγωγή της μελωδίας θα τυπώνεται στην οθόνη του υπολογιστή η νότα, η συχνότητά της, η χρονική της αξία, καθώς και η διάρκειά της σε msec. Αν δε γραφεί αυτή η εντολή η μελωδία θα αναπαράγεται, χωρίς να δείχνει στην οθόνη τι παίζεται κάθε χρονική στιγμή. Με την εντολή **end=" "** που υπάρχει στην πρώτη **print** αποτρέπουμε την αλλαγή γραμμής για τη δεύτερη **print** και έτσι δίνουμε ουσιαστικά εντολή να τυπωθεί δίπλα.

**Γραμμή 26:** Ορίζουμε το **Beer** να καθορίζεται από τη συχνότητα και τη χρονική διάρκεια.

**Γραμμή 27:** Εντολή αναπαραγωγής.

Στη συνέχεια πατάμε το βελάκι πлагστη γραμμή εργασιών και καλή ακρόαση , έστω και με φτωχό ήχο.....

**Jingle Bells**  
PianoTels.com Traditional

Jin - gle bells, jin - gle bells, jin - gle all the way! Oh, what fun it is to ride in a  
one horse op - en sleigh. Hey! Jin - gle bells, jin - gle bells, jin - gle all the  
way! Oh, what fun it is to ride in a one horse op - en sleigh!

12

Κάθε οκτάβα έχει δώδεκα διαφορετικούς φθόγγους. Μπορούμε να υπολογίσουμε τη συχνότητα κάθε επόμενου φθόγγου από τον προηγούμενο, πολλαπλασιάζοντας με  $\sqrt[12]{2}=1,059446309435920953$ . Στρογγυλοποιώντας το αποτέλεσμα παίρνουμε τις συχνότητες των παρακάτω δώδεκα φθόγγων:

Do: 262Hz

Do#: 277Hz

Re: 294Hz

Re#: 311Hz

Mi: 330Hz

Fa: 349Hz

Fa#: 370Hz

Sol: 392Hz

Sol#: 415Hz

La: 440Hz

La#: 466Hz

Si: 494Hz

Do1: 523Hz

Παρατηρήστε ότι η Do1 στην ψηλότερη οκτάβα είναι διπλάσιας συχνότητας από τη Do. Μπορούμε πολλαπλασιάζοντας επί 2 ή διορώντας δια 2 τη συχνότητα κάθε νότας, να οδηγηθούμε σε υψηλότερες ή χαμηλότερες οκτάβες αντίστοιχα.

Τώρα μπορούμε να προγραμματίσουμε και να αναπαράγουμε και άλλες μελωδίες και να περάσουμε ευχάριστα και δημιουργικά...