

music-player-framework-in-c

#class files

*.class

#executable files

*.exe

#output files from mac

*.out

CODE

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <stdbool.h>

#include <ctype.h>

#include <math.h>

#ifdef _WIN32

#include <windows.h>

#else

#include <unistd.h>

#endif

#define MAX_SONG_POOL 120

#define MAX_STRING_SIZE 75

```
// Data Structure for an individual song
```

```
typedef struct Song *songptr;
```

```
typedef struct Song
```

```
{
```

```
    char title[MAX_STRING_SIZE];
```

```
    char album[MAX_STRING_SIZE];
```

```
    char uri[MAX_STRING_SIZE];
```

```
    char id[MAX_STRING_SIZE];
```

```
    short int year;
```

```
    double duration;
```

```
} Song;
```

```
songptr song_pool[MAX_SONG_POOL];
```

```
int pool_insert_index = 0;
```

```
typedef struct PlaylistNode *node_ptr;
```

```
typedef struct PlaylistNode
```

```
{
```

```
    songptr song;
```

```
    node_ptr prev_song;
```

```
    node_ptr next_song;
```

```
} PlaylistNode;
```

```
node_ptr header_node = NULL, now_playing = NULL;
```

```
//Function to check if the song pool is empty
```

```
bool is_pool_empty()
```

```
{  
    return song_pool[0] == NULL;  
}
```

```
//Function to check if playlist exists
```

```
bool does_playlist_exist()
```

```
{  
    return !(header_node == NULL);  
}
```

```
void play_next_song()
```

```
{  
    if (does_playlist_exist() && now_playing != NULL)  
    {  
        if (now_playing->next_song != NULL)  
            now_playing = now_playing->next_song;  
        else  
            printf("REACHED END OF PLAYLIST");  
    }  
    else  
    {  
        printf("NO SONG ADDED TO PLAYLIST");  
    }  
}
```

```
    }  
}
```

```
void play_prev_song()
```

```
{  
    if (does_playlist_exist() && now_playing != NULL)  
    {  
        if (now_playing->prev_song != NULL)  
            now_playing = now_playing->prev_song;  
        else  
            printf("REACHED START OF PLAYLIST");  
    }  
    else  
    {  
        printf("NO SONG ADDED TO PLAYLIST");  
    }  
}
```

```
void show_song_details()
```

```
{  
    if (now_playing == NULL || now_playing->song == NULL)  
    {  
        printf("                CREATE A PLAYLIST FIRST\n");  
    }  
    else
```

```

{

printf("\n-----\n");

printf("                DETAILED OVERVIEW\n");

printf("-----\n");

printf("\t TITLE   |  %s\n", now_playing->song->title);

printf("\t ALBUM    |  %s\n", now_playing->song->album);

printf("\t YEAR     |  %d\n", now_playing->song->year);

printf("\t DURATION |  %0.2lf minutes\n", now_playing->song->duration);

printf("\tSPOTIFY URI |  %s\n", now_playing->song->uri);

printf("   Paste the above URI in your browser to play the song on Spotify\n");

printf("-----\n");

}

char leave[MAX_STRING_SIZE];

printf("<<<<< Enter any input to return to MAIN MENU\n");

if (scanf("%s", leave))

{

    system("clear");

    return;

}

}

void main_menu()

{

printf("\n-----\n");

printf("                MAIN MENU\n");

```

```

printf("-----\n");

if (now_playing == NULL || now_playing->song == NULL)
{
    printf("          NO TRACK SELECTED\n");
}

else
{
    printf("  NOW    | %s\n", now_playing->song->title);
    printf("  PLAYING | %0.2lf minutes\n", now_playing->song->duration);
}

printf("-----\n");

printf("  #   | Action   \n");

printf("-----\n");

printf("  1   | Create a new song\n");
printf("  2   | Display all available songs\n");
if (does_playlist_exist())
    printf("  3   | Delete playlist\n");
else
    printf("  3   | Create a new playlist\n");
printf("  4   | Add a song to the playlist\n");
printf("  5   | Show playlist\n");
printf("  6   | Play previous track\n");
printf("  7   | Play next track\n");

if (now_playing != NULL && now_playing->song != NULL)

```

```

        printf(" 8 | Show more information about the song playing\n");

printf(" -1 | Exit music player\n");

printf("-----\n");

printf("          Enter your choice below\n");
}

void show_all_songs_raw()
{
    for (int i = 0; i < MAX_SONG_POOL && song_pool[i] != NULL; i++)

        printf("%-1d %-3s %-3s %-3d %0.2lfmin\n", (i + 1), song_pool[i]->title,
song_pool[i]->album, song_pool[i]->year, song_pool[i]->duration);
}

void show_playlist()
{
    if (does_playlist_exist())
    {
        printf("\n-----\n");

        printf("          YOUR PLAYLIST\n");

        printf("-----\n");

        printf(" Title                      | Duration\n");

        printf("-----\n");

        node_ptr current = header_node;

        while (current != NULL && current->song != NULL)
        {
            node_ptr next = current->next_song;

```

```

        printf(" %-48.48s | %2.2lf min\n", current->song->title, current->song->duration);

        current = next;

    }

    printf("-----");
}

else
{
    printf("\nYou haven't created a playlist yet.\n");
}
}

void pagewise_song_display(int step)
{
    printf("\n-----\n");

    printf("                SONGS LIST\n");

    printf("-----\n");

    printf(" # | Title | Duration\n");

    printf("-----\n");

    for (int i = step - 10; i < step && song_pool[i] != NULL; i++)
    {
        printf(" %2d | %-48.48s | %2.2lf min\n", (i + 1), song_pool[i]->title, song_pool[i]->duration);
    }

    printf("-----\n");
}

```



```
printf("[Enter -2 to go to prev page] | Page %d | [Enter -1 to go to next page]\n",  
((int)step / 10));
```

```
printf("-----\n");
```

```
printf(" <<<< Enter 0 to go back to main menu.\n");
```

```
printf("-----\n");
```

```
printf("                Enter your choice below\n");
```

```
}
```

```
int song_selector()
```

```
{
```

```
    int song_number = -1;
```

```
    int step = 10;
```

```
    while (song_number < 0 || song_number > pool_insert_index)
```

```
    {
```

```
        pagewise_song_display(step);
```

```
        char input[MAX_STRING_SIZE];
```

```
        scanf("%s", input);
```

```
        if (!sscanf(input, "%d", &song_number))
```

```
        {
```

```
            system("clear");
```

```
            printf("You seem to have entered an invalid input.\nReturning to main menu.");
```

```
            return 0;
```

```
        }
```

```
if (song_number == -2)
{
    if (step > 10)
    {
        step -= 10;
        system("clear");
    }
    else
    {
        system("clear");
        printf("\nThis itself is the first page");
    }
}

else if (song_number == -1)
{
    if (step < ceil(pool_insert_index / 10.0) * 10)
    {
        step += 10;
        system("clear");
    }
    else
    {
        system("clear");
        printf("\nThis is the last page");
    }
}
```

```

    }

    else if (song_number < -2 || song_number > pool_insert_index)

    {

        system("clear");

        printf("\nENTER A VALID CHOICE\n");

    }

}

return song_number;

}

```

Song *createSong(const char *title, const char *album, const short int year, const double duration, const char *uri, const char *id)

```

{

    Song *temp = malloc(sizeof(Song));

    strcpy(temp->title, title);

    strcpy(temp->album, album);

    strcpy(temp->uri, uri);

    strcpy(temp->id, id);

    temp->year = year;

    temp->duration = duration;

    pool_insert_index++;

    return temp;

}

```

//Allocates memory for a new playlist and prompts user to add a song to it

```

void create_playlist()
{
    int song_number = 0, i = 0;

    node_ptr temp = (node_ptr)malloc(sizeof(PlaylistNode));

    printf("\nPick the song you want to add to your new playlist\n");

    song_number = song_selector();

    if (song_number > 0 && song_number <= 100)
    {
        temp->song = song_pool[song_number - 1];

        temp->prev_song = NULL;

        temp->next_song = NULL;

        header_node = temp;

        now_playing = temp;

        system("clear");

        printf("---\n%s has been added to your new playlist.\n---", song_pool[song_number - 1]->title);

    }

    else if (song_number == 0)
    {
        system("clear");

    }

    else
    {
        printf("\nThere was a problem while handling your request, try again.\n");
    }
}

```

```
}  
  
}
```

//Prompts user to pick a song from the pool and add it to the existing playlist

```
void add_to_playlist()
```

```
{
```

```
    int song_number = 0;
```

```
    int step = 10;
```

```
    printf("\nPick the song you want to add\n");
```

```
    song_number = song_selector();
```

```
    if (song_number > 0 && song_number <= pool_insert_index)
```

```
    {
```

```
        node_ptr new_node = (node_ptr)malloc(sizeof(PlaylistNode));
```

```
        node_ptr last = header_node;
```

```
        new_node->song = song_pool[song_number - 1];
```

```
        new_node->next_song = NULL;
```

```
        /* 4. If the Linked List is empty, then make the new
```

```
        node as head */
```

```
        if (header_node == NULL)
```

```
        {
```

```
            new_node->prev_song = NULL;
```

```
            header_node = new_node;
```

```
            now_playing = new_node;
```

```
        }
```

```

else

{

    //Traverse till the last node

    while (last->next_song != NULL)

        last = last->next_song;


    last->next_song = new_node;

    new_node->prev_song = last;

}

system("clear");

printf("%s has been added to your playlist.\n", song_pool[song_number - 1]->title);

}

else if (song_number == 0)

{

    system("clear");

}

else

{

    printf("\nThere was a problem while handling your request.\n");

}

}

```

/** CSV Parser that maps a data set of upto songs onto the song pool

Songs must be present in the root directory at rawdata.csv

*/

```

void readFromCSV()

{

    FILE *file = fopen("rawdata.csv", "r");

    char line[1000];

    int i = 0;


    if (file == NULL)

    {

        perror("*****\nUnable to load songs from data source");

        printf("Please check if rawdata.csv exists at the root directory of the application.\n");

        printf("You can still enter songs manually.\n*****\n");

        printf("Error Message");

    }

    else

    {

        char songtitle[MAX_STRING_SIZE];

        char releaseDate[MAX_STRING_SIZE];

        int durationms;

        double duration = 0.0;

        char id[MAX_STRING_SIZE];

        char album[MAX_STRING_SIZE];

        char uri[MAX_STRING_SIZE];

        int year;

        while (fgets(line, sizeof(line), file) && pool_insert_index < MAX_SONG_POOL)

        {

```

```

char *token;

token = strtok(line, ","); //tokenizes current line with delimiter ',' and returns the
first substring

if (token == NULL)

    continue;

strcpy(songtitle, token);

token = strtok(NULL, ","); // get the next token

strncpy(releaseDate, token + 1, 4);

sscanf(releaseDate, "%d", &year);

token = strtok(NULL, ",");

char temp[MAX_STRING_SIZE];

strncpy(temp, token + 1, 6);

sscanf(temp, "%d", &durationms);

duration = durationms / 60000.0;

token = strtok(NULL, ","); // get the next token

strcpy(id, token);

token = strtok(NULL, ","); // get the next token

strcpy(album, token);

token = strtok(NULL, ","); // get the next token

strcpy(uri, token);

token = strtok(NULL, ","); // get the next token

//printf("Song read : %s %s\n", songname, durationm);

if (i == 0)

{

    i++;

    continue;

```



```

    }

    song_pool[i - 1] = createSong(songtitle, album, year, duration, uri, id);

    i++;

}

}

//printf("%d Songs added.\n\n", i);

fclose(file);

}

```

//Function that allows the user to create a song of their own

```

void user_song_input()
{
    int add_another = 1;

    char songname[MAX_STRING_SIZE];

    int year;

    int durationms;

    char id[MAX_STRING_SIZE];

    char album[MAX_STRING_SIZE];

    char uri[MAX_STRING_SIZE] = "spotify:track:";

    double duration = 0.0;

    while (add_another && pool_insert_index < MAX_SONG_POOL)
    {
        system("clear");

```

```
printf("\nEnter the details of the song you want to create.\n[ Don't give spaces  
between each word, use _ instead\n");
```

```
printf("Title: ");  
  
scanf("%s", songname);  
  
printf("Album Name: ");  
  
scanf("%s", album);  
  
printf("Year of release: ");  
  
if (!(scanf("%d", &year)))  
{  
  
    printf("INVALID INPUT ENTERED\n");  
  
    sleep(1);  
  
    break;  
  
}  
  
printf("Duration(in s): ");  
  
if (!(scanf("%d", &durationms)))  
{  
  
    printf("INVALID INPUT ENTERED\n");  
  
    sleep(1);  
  
    break;  
  
}  
  
duration = durationms / 60;  
  
printf("Song ID: ");  
  
scanf("%s", id);  
  
strcat(uri, id);
```

```

    song_pool[pool_insert_index] = createSong(songname, album, year, duration, uri,
id);

    printf("\nThe song %s has been created.\n\n", songname);

    printf("\nDo you want to create another song?\n(Enter 1 for yes and 0 for no) : ");

    scanf("%d", &add_another);

}

system("clear");

}

```

//Helper function to delete the playlist created and free all resources taken up by PlaylistNodes

```

bool delete_playlist()

{

    if (header_node != NULL)

    {

        node_ptr current = header_node;

        while (current != NULL)

        {

            node_ptr next = current->next_song;

            free(current);

            current = next;

        }

        header_node = NULL;

        now_playing = NULL;

        return true;

    }

```

```

        return false;
    }

//Frees resources that had been dynamically allocated
void free_all_memory()
{
    delete_playlist();

    for (int i = 0; i < MAX_SONG_POOL && song_pool[i] != NULL; i++)
        free(song_pool[i]);

    printf("\n---END---");

    //Generate a 2 second delay

    sleep(2);
}

int main()
{
    int userChoice = 0;

    char *terminate = "X";

    int wrong_choice_count = 0;

    system("clear");

    readFromCSV();

    while (userChoice != -1)
    {
        switch (userChoice)

```

```
{  
  
case 0:  
  
{ //Show menu options  
  
    main_menu();  
  
    break;  
  
}  
  
case 1:  
  
{  
  
    system("clear");  
  
    if (pool_insert_index >= MAX_SONG_POOL)  
  
    {  
  
        printf("Cannot add more songs.\n");  
  
    }  
  
  
  
    user_song_input();  
  
    break;  
  
}  
  
case 2:  
  
{  
  
    system("clear");  
  
    //show_all_songs();  
  
    int user_song_selection = song_selector();  
  
    system("clear");  
  
    break;  
  
}
```

case 3:

```
{  
    system("clear");  
    if (does_playlist_exist())  
    {  
        printf("---");  
        if (delete_playlist())  
            printf("\nThe playlist was successfully deleted.\n");  
        else  
            printf("\nYou haven't created a playlist yet. Nothing to delete\n");  
  
        printf("---");  
    }  
    else  
        create_playlist();  
    break;  
}
```

case 4:

```
{  
    system("clear");  
    add_to_playlist();  
    break;  
}
```

case 5:

```
{
```

```
        system("clear");

        show_playlist();

        break;
    }

    case 6:

    {

        system("clear");

        play_prev_song();

        break;
    }

    case 7:

    {

        system("clear");

        play_next_song();

        break;
    }

    case 8:

    {

        system("clear");

        show_song_details();

        break;
    }

    default:

    {
```

```
if (wrong_choice_count == 3)
{
    system("clear");

    printf("Please enter a valid option or the program will terminate.\n");
}
if (wrong_choice_count > 3)
{
    system("clear");

    printf("Sorry you have exceeded the maximum number of retries,
terminating..");

    free_all_memory();

    exit(1);
}
else
{
    system("clear");

    printf("Please enter a valid option from the main menu below\n");
}

wrong_choice_count++;

break;
}
}

if (userChoice != 0)
{
    main_menu();
}
```



```
char input[MAX_STRING_SIZE];

scanf("%s", input);

if (!sscanf(input, "%d", &userChoice))
{
    system("clear");

    printf("----\nINVALID INPUT\n");

    userChoice = 0;

    wrong_choice_count++;

    if (wrong_choice_count > 3)
    {
        printf("Sorry you have exceeded the maximum number of retries,
terminating..");

        free_all_memory();

        exit(1);
    }

    continue;
}

free_all_memory();

return 0;
}
```

Literature Survey

The idea of implementing data structures in a music player was influenced by a research paper titled - "New Implementation of Digital Jukebox in Data Structure By Using C++ Programming Language", published by Manigandan Raamanathan from the Asia Pacific University, Malaysia. The paper was published by the International Journal of Advanced Research in Science and Engineering in Vol. No.6, Issue No.4 April 2017. The paper demonstrates an implementation of a "Digital Jukebox" using linked lists, queues and stacks in C++.

We have recreated the experience of a simple music player that allows users to load a pool of songs from local storage and play tracks, create playlists and interact with it in a simple manner. The linked list data structure allows us to scrub through the entire playlist as per user's instructions. To obtain a collection of songs we used the Spotify Developers API to save a CSV file of upto 100 songs' metadata which we parse in our application to simulate the loading of songs from local storage. Our application features a simple menu based interface that allows all users to navigate with ease.

Screenshots

```
-----
                                MAIN MENU
-----
NOW PLAYING | "Levitating (feat. DaBaby)"
              | 3.38 minutes
-----
# | Action
-----
1 | Create a new song
2 | Display all available songs
3 | Delete playlist
4 | Add a song to the playlist
5 | Show playlist
6 | Play previous track
7 | Play next track
8 | Show more information about the song playing
-1 | Exit music player
-----
Enter your choice below
```

(MAIN MENU)

SONGS LIST		
#	Title	Duration
31	"Goosebumps"	2.72 min
32	"Wildboy"	2.00 min
33	"MEGATRON"	3.18 min
34	"Savage Remix (feat. Beyonc -)"	4.03 min
35	"i miss u (with Au/Ra)"	2.87 min
36	"Music To Be Murdered By"	3.51 min
37	"Desce Pro Play (PA PA PA)"	2.81 min
38	"HIGHEST IN THE ROOM"	2.93 min
39	"Kill Me Better"	3.23 min
40	"One More Dance (with Alida)"	2.42 min
[Enter -2 to go to prev page] Page 4 [Enter -1 to go to next page]		
<<<< Enter 0 to go back to main menu.		
Enter your choice below		

(SONGS LIST)

```
Enter the details of the song you want to create.
[ Don't give spaces between each word, use _ instead
Title: Dynamite
Album Name: Dynamite_Single_By_BTS
Year of release: 2020
Duration(in s): 199
Song ID: u7823bbfsbj21bts

The song Dynamite has been created.

Do you want to create another song?
(Enter 1 for yes and 0 for no) :
```

(ADD A NEW SONG)

DETAILED OVERVIEW	
TITLE	"Levitating (feat. DaBaby)"
ALBUM	"Levitating (feat. DaBaby)"
YEAR	2020
DURATION	3.38 minutes
SPOTIFY URI	"spotify:track:463CkQjx2Zk1yXoBuierM9"
Paste the above URI in your browser to play the song on Spotify	
<<<<< Enter any input to return to MAIN MENU	

(MORE INFO ABOUT SONG)

YOUR PLAYLIST	
Title	Duration
"Levitating (feat. DaBaby)"	3.38 min
"Ride It"	2.63 min
"Hollywood's Bleeding"	3.83 min
"Future Nostalgia"	3.23 min
"HIGHEST IN THE ROOM"	2.93 min
"Se Te Nota (with Guaynaa)"	2.60 min
"BOOM"	2.99 min
"All We Got (feat. KIDDO) [Ofenbach Remix]"	2.16 min

	track_album_release_date	track_duration_ms	track_id	track_name	track_uri	
2	Don't Start Now	2019-10-31	183290	6WrI0LAC5M1Rw2MnX2ZvEg	Don't Start Now	spotify:track:6WrI0LAC5M1Rw2MnX2ZvEg
3	Mood (feat. iann dior)	2020-07-24	140525	3tjFYV6RSFtuktYl3ZtYcq	Mood (feat. iann dior)	spotify:track:3tjFYV6RSFtuktYl3ZtYcq
4	Positions	2020-10-30	172324	35mvY5S1H3J2QZyna3TFe0	positions	spotify:track:35mvY5S1H3J2QZyna3TFe0
5	Breaking Me	2019-12-19	166793	3H7ihDc1dqLriiWxwsc2po	Breaking Me	spotify:track:3H7ihDc1dqLriiWxwsc2po
6	Blueberry Faygo	2020-02-07	162546	6wJYhPfqk3KGhHRG76WzOh	Blueberry Faygo	spotify:track:6wJYhPfqk3KGhHRG76WzOh
7	Ride It	2019-07-26	157605	2tnVG71enUj3lcnFN6kZ	Ride It	spotify:track:2tnVG71enUj3lcnFN6kZ
8	Lonely	2020-01-24	190955	2ZEq4HT450Ye9lFGPTl9qV	Lonely	spotify:track:2ZEq4HT450Ye9lFGPTl9qV
9	BLAME IT ON BABY	2020-04-17	181733	7ytR5pFWmSjzHJleQkgog4	ROCKSTAR (feat. Roddy Ricch)	spotify:track:7ytR5pFWmSjzHJleQkgog4
10	Please Excuse Me For Being Antisocial	2019-12-06	196652	0nbXyq5TXYPcO7pr3N8S4l	The Box	spotify:track:0nbXyq5TXYPcO7pr3N8S4l
11	Future Nostalgia	2020-03-27	193829	3AzjcOeAmA57TIOr9zF1ZW	Physical	spotify:track:3AzjcOeAmA57TIOr9zF1ZW
12	We Love You Tecca	2019-08-30	131240	1lOe9qE0vR9zwWQAOk6CoO	Ransom	spotify:track:1lOe9qE0vR9zwWQAOk6CoO
13	Hollywood's Bleeding	2019-09-06	149546	7xQAfvXzm3AkraOtGPWIZg	Wow.	spotify:track:7xQAfvXzm3AkraOtGPWIZg

	track_album_release_date	track_duration_ms	track_id	track_name	track_uri	
1 4	Roses (Imanbek Remix)	2019-10-09	17 68 40	24Yi9hE7 8yPEbZ4k xyoXAI	Roses - Imanbek Remix	spotify:track:24Yi9hE78yPEbZ4kxyoXAI
1 5	TKN (feat. Travis Scott)	2020-05-28	12 93 70	4w47S36 wQGBhG g073q3nt 7	TKN (feat. Travis Scott)	spotify:track:4w47S36wQGBhGg073q3nt7
1 6	Elyanna	2020-02-07	17 99 88	0fRlZjNaV xFqwoLg 8rfQX5	Ana Lahale	spotify:track:0fRlZjNaVxFqwoLg8rfQX5
1 7	Head & Heart (feat. MNEK)	2020-07-03	16 60 28	6cx06DFP PHchuUA cTxznu9	Head & Heart (feat. MNEK)	spotify:track:6cx06DFPPHchuUAcTxznu9
1 8	Legends Never Die	2020-07-10	20 54 84	2Y0wPrP QBrGhoL n14xRYC G	Come & Go (with Marshmello)	spotify:track:2Y0wPrPQBrGhoLn14xRYCG
1 9	Levitating (feat. DaBaby)	2020-10-01	20 30 64	463CkQjx 2Zk1yXoB uierM9	Levitating (feat. DaBaby)	spotify:track:463CkQjx2Zk1yXoBuierM9
2 0	High On Life (feat. Bonn)	2018-07-29	23 07 61	4ut5G4rg B1ClpMT Mfjoluy	High On Life (feat. Bonn)	spotify:track:4ut5G4rgB1ClpMTMfjoluy
2 1	Why Do You Lie To Me (feat. Lil Baby)	2020-08-28	17 12 90	1Mu0qs9 DQ8Ofhi PvHxZM MM	Why Do You Lie To Me (feat. Lil Baby)	spotify:track:1Mu0qs9DQ8OfhiPvHxZMMM
2 2	Lemonade (feat. Gunna, Don Toliver & NAV)	2020-08-14	19 54 28	02kDW37 9Yfd5Pz W5A6vuG t	Lemonade	spotify:track:02kDW379Yfd5PzW5A6vuGt
2 3	Evolve	2017-06-23	20 43 46	0pqnGHJ pmpxLKif KRmU6W P	Believer	spotify:track:0pqnGHJpmpxLKifKRmU6WP
2 4	Paradise	2020-10-30	16 79 03	6ft4hAq6 yde8jPZY 2i5zLr	Paradise (feat. Dermot Kennedy)	spotify:track:6ft4hAq6yde8jPZY2i5zLr
2 5	JACKBOYS	2019-12-27	15 77 12	6gi6y1xw mVszDW	OUT WEST (feat. Young Thug)	spotify:track:6gi6y1xwmVsz

	track_album_release_date	track_duration_ms	track_id	track_name	track_uri	
				kUqab1qw		DWkUqab1qw
26	Too Much (feat. Usher)	2020-10-23	165704	39umHdGKfnpjY4TFtnoQ51	Too Much (feat. Usher)	spotify:track:39umHdGKfnpjY4TFtnoQ51
27	ASCEND (Remixes)	2020-05-15	247384	42iaZygYmdyZomAtvPoxG1	Good Things Fall Apart (with Jon Bellion) [Tiësto's Big Room Remix]	spotify:track:42iaZygYmdyZomAtvPoxG1
28	Prisoner (feat. Dua Lipa)	2020-11-20	169333	5JqZ3oqF00jkT81foAFvqg	Prisoner (feat. Dua Lipa)	spotify:track:5JqZ3oqF00jkT81foAFvqg
29	Higher Ground (feat. John Martin)	2020-05-14	206279	0rohJsT6NWsThpukt0Xxdc	Higher Ground (feat. John Martin)	spotify:track:0rohJsT6NWsThpukt0Xxdc
30	More Than You Know	2017-07-28	203000	6h5PAsRni4IRlxWr6uDPTP	More Than You Know	spotify:track:6h5PAsRni4IRlxWr6uDPTP
31	Pretty Please	2020-09-04	145792	5Rse8lf81NbzxL1ARjHOTB	Pretty Please	spotify:track:5Rse8lf81NbzxL1ARjHOTB
32	Goosebumps	2020-06-12	163208	20McUFi8KCIVdNdLrkTDuZ	Goosebumps	spotify:track:20McUFi8KCIVdNdLrkTDuZ
33	Wildboy	2019-05-31	120133	7rvyVWja33WG9R97oeJAJx	Envy Me	spotify:track:7rvyVWja33WG9R97oeJAJx
34	MEGATRON	2019-06-21	190746	1ZDmxwg7LywIx68b7SsJjj	MEGATRON	spotify:track:1ZDmxwg7LywIx68b7SsJjj
35	Savage Remix (feat. Beyoncé)	2020-04-29	242000	5v4GgrXPMghOnBBLmveLac	Savage Remix (feat. Beyoncé)	spotify:track:5v4GgrXPMghOnBBLmveLac
36	i miss u (with Au/Ra)	2020-10-09	172143	7CH8J4uIT49UfZwSDSkSZA	i miss u (with Au/Ra)	spotify:track:7CH8J4uIT49UfZwSDSkSZA
37	Music To Be Murdered By	2020-01-17	210800	7FIWs0pqAYbP91WWM0vITQ	Godzilla (feat. Juice WRLD)	spotify:track:7FIWs0pqAYbP91WWM0vITQ

	track_album_release_date	track_duration_ms	track_id	track_name	track_uri	
3 8	Desce Pro Play (PA PA PA)	2020-06-26	16 85 07	4zGtfOgl DBfVUyp abBwljQ	Desce Pro Play (PA PA PA)	spotify:track:4 zGtfOglDBfV UypabBwljQ
3 9	HIGHEST IN THE ROOM	2019-10-04	17 57 20	3eekarcy 7kvN4yt5 ZFzltW	HIGHEST IN THE ROOM	spotify:track:3 eekarcy7kvN4 yt5ZFzltW
4 0	Kill Me Better	2020-10-30	19 35 68	7MtJ0dU 4m3F6JQ G9qdyAP k	Kill Me Better	spotify:track:7 MtJ0dU4m3F 6JQG9qdyAPk
4 1	One More Dance (with Alida)	2020-11-13	14 53 20	2aOK6TR Yn4DnelB HJOAQ4r	One More Dance (with Alida)	spotify:track:2 aOK6TRYn4D nelBHJOAQ4r
4 2	Shoot For The Stars Aim For The Moon	2020-07-03	19 04 76	0PvFJman yNQMsel FrU708S	For The Night (feat. Lil Baby & DaBaby)	spotify:track:0 PvFJmanyNQ MselFrU708S
4 3	YE YE	2020-07-31	18 29 41	6CVSoFx NZtg0Rp RNTAYop U	YE YE	spotify:track:6 CVSoFxNZtg0 RpRNTAYopU
4 4	Head Shoulders Knees & Toes (feat. Norma Jean Martine)	2020-05-08	15 69 60	14ngWW xvUSnIM XgF6rzSk 1	Head Shoulders Knees & Toes (feat. Norma Jean Martine)	spotify:track:1 4ngWWxvUS nIMXgF6rzSk 1
4 5	Body (Dzeko Remix)	2018-08-27	16 57 14	4RgDgao hkG08f63 0ZT9QKc	Body - Dzeko Remix	spotify:track:4 RgDgaohkG0 8f630ZT9QKc
4 6	Nobody	2020-10-30	14 39 52	7GiozRo Mk95aFl1 WbrDdjX	Nobody	spotify:track:7 GiozRoMk95a Fl1WbrDdjX
4 7	Pump It Up	2019-09-13	15 06 35	0GiwV6v3 AgJfdu59 tj719Y	Pump It Up	spotify:track:0 GiwV6v3AgJf du59tj719Y
4 8	Fire (with Elderbrook)	2020-09-04	16 39 68	5HFPK8m cO6BGPZ NeZPZI5L	Fire (with Elderbrook)	spotify:track:5 HFPK8mcO6B GPZNeZPZI5L
4 9	Goddess	2020-05-29	17 74 99	4KloBpLd HX7nERPf DROJWR	Goddess	spotify:track:4 KloBpLdHX7n ERPfDROJWR

	track_album_release_date	track_duration_ms	track_id	track_name	track_uri	
50	Se Te Nota (with Guaynaa)	2020-09-02	155825	11EnQRgRMJwMAesfkB5pnu	Se Te Nota (with Guaynaa)	spotify:track:11EnQRgRMJwMAesfkB5pnu
51	Life Is Good (feat. Drake)	2020-01-10	237735	5yY9IUy8nbvjM1Uyo1Uqoc	Life Is Good (feat. Drake)	spotify:track:5yY9IUy8nbvjM1Uyo1Uqoc
52	Project Dreams	2019-01-25	167898	6GSkLS2y4ZnnZsuRAj975K	Project Dreams	spotify:track:6GSkLS2y4ZnnZsuRAj975K
53	Like Gold	2020-10-30	176388	5seWZTjT0U8a7cy3Y5NJQ9	Like Gold	spotify:track:5seWZTjT0U8a7cy3Y5NJQ9
54	Say My Name	2020-07-24	159535	3GYlQmwfBDo7imFyGPhrl7	Say My Name	spotify:track:3GYlQmwfBDo7imFyGPhrl7
55	Rust to Gold	2018-02-16	226089	3QA66YGqHvdSthvEAhGmMH	Rust to Gold	spotify:track:3QA66YGqHvdSthvEAhGmMH
56	Sad Songs (feat. L Devine)	2020-10-30	191612	3JsmrruMp8AWcbEZ4342JY	Sad Songs (feat. L Devine)	spotify:track:3JsmrruMp8AWcbEZ4342JY
57	ASTROWORLD	2018-08-03	312820	2xLMifQCjDGFmkHkpNLD9h	SICKO MODE	spotify:track:2xLMifQCjDGFmkHkpNLD9h
58	Euphoric Sad Songs	2020-11-20	197573	3DTqHfTGj1c6y2gDXsTez4	Regardless	spotify:track:3DTqHfTGj1c6y2gDXsTez4
59	The Business	2020-09-16	164000	6f3Slt0GbA2bPZlZ0aIFXN	The Business	spotify:track:6f3Slt0GbA2bPZlZ0aIFXN
60	POPSTAR (feat. Drake)	2020-07-17	200221	6EDO9iITtwNv6waLwa1UUq	POPSTAR (feat. Drake)	spotify:track:6EDO9iITtwNv6waLwa1UUq
61	Hollywood's Bleeding	2019-09-06	196760	0Xek5rqai2jcOWCYWJfVCF	Enemies (feat. DaBaby)	spotify:track:0Xek5rqai2jcOWCYWJfVCF

	track_album_release_date	track_duration_ms	track_id	track_name	track_uri	
6 2	Hoodie SZN	2018-12-21	189486	1wJRveJZLSb1rjhnUHQiv6	Swervin (feat. 6ix9ine)	spotify:track:1wJRveJZLSb1rjhnUHQiv6
6 3	Nights Like This (PBH & Jack Remix)	2020-08-14	173333	24BBzfOsyK6YELZSo0ILNG	Nights Like This - PBH & Jack Remix	spotify:track:24BBzfOsyK6YELZSo0ILNG
6 4	Life After You	2020-07-31	188081	2c1BSmrSeBEY9FxyzSiFWWh	Life After You	spotify:track:2c1BSmrSeBEY9FxyzSiFWWh
6 5	Without You	2020-05-01	190729	3iQNkoYyr2SffgYiTpE3v5	Without You	spotify:track:3iQNkoYyr2SffgYiTpE3v5
6 6	Hey Baby (feat. Gia Koka)	2020-08-28	177545	3bNIIJBpwgFNnFuSjH6Uh	Hey Baby (feat. Gia Koka)	spotify:track:3bNIIJBpwgFNnFuSjH6Uh
6 7	Delali	2020-05-12	183529	1yGPoS9sQvB8TPfa2wEv9w	Delali	spotify:track:1yGPoS9sQvB8TPfa2wEv9w
6 8	No Service In The Hills (feat. Trippie Redd, Blackbear, PRINCE\$\$ ROSIE) [NGHTMRE Remix]	2020-04-03	232105	0KaaI4pEUw3hmnVHW3R5IW	No Service In The Hills (feat. Trippie Redd, Blackbear, PRINCE\$\$ ROSIE) - NGHTMRE Remix	spotify:track:0KaaI4pEUw3hmnVHW3R5IW
6 9	Dakiti	2020-10-30	205090	47EiUVWUp4C9fGccaPuUCS	Dakiti	spotify:track:47EiUVWUp4C9fGccaPuUCS
7 0	BOOM	2018-01-26	179512	6iC7vapdKp0XIKhxOkp8QJ	BOOM	spotify:track:6iC7vapdKp0XIKhxOkp8QJ
7 1	Featuring Ty Dolla \$ign	2020-10-23	143375	5IUtvfNvOyVYZUa6AJFrnP	Spicy (feat. Post Malone)	spotify:track:5IUtvfNvOyVYZUa6AJFrnP
7 2	Heart On My Sleeve	2020-06-26	209062	32hvlKjjOhc15tnVmNk4wE	Heart On My Sleeve	spotify:track:32hvlKjjOhc15tnVmNk4wE
7 3	Coño (Henry Fong Remix)	2020-09-11	139000	6m5JQ8osGRHPqoRRbRiEE	Coño - Henry Fong Remix	spotify:track:6m5JQ8osGRHPqoRRbRiEE

	track_album_release_date	track_duration_ms	track_id	track_name	track_uri	
7 4	Wired (with Ella Eyre)	2020-10-09	202652	7k3uKOLzMCvYM00E3QZTSq	Wired (with Ella Eyre)	spotify:track:7k3uKOLzMCvYM00E3QZTSq
7 5	Sweet Action	2020-03-13	139741	1jaTQ3nqY3oAAYyCTblvnM	WHATS POPPIN	spotify:track:1jaTQ3nqY3oAAYyCTblvnM
7 6	Up All Night (feat. Afrojack)	2020-10-09	160906	5k1ulPpsHmnbWFr4nbpdyb	Up All Night (feat. Afrojack)	spotify:track:5k1ulPpsHmnbWFr4nbpdyb
7 7	Hold Me Back	2019-04-26	223107	5C5FqRfbXF9oY4HSRYyqn3	Hold Me Back	spotify:track:5C5FqRfbXF9oY4HSRYyqn3
7 8	Dreamer (Remixes)	2018-02-02	250809	6Bz6Rsp8GyeK4yL7APYYX3	Dreamer - Matisse & Sadko Remix	spotify:track:6Bz6Rsp8GyeK4yL7APYYX3
7 9	Laugh Now Cry Later (feat. Lil Durk)	2020-08-14	261492	2SAqBLGA283SUIwJ3xOUVl	Laugh Now Cry Later (feat. Lil Durk)	spotify:track:2SAqBLGA283SUIwJ3xOUVl
8 0	All We Got (feat. KIDDO) [Ofenbach Remix]	2020-11-13	129814	2rUsJXt4NeeDiIU6sBvrxo	All We Got (feat. KIDDO) - Ofenbach Remix	spotify:track:2rUsJXt4NeeDiIU6sBvrxo
8 1	Creep (mint Singles)	2020-03-20	161190	3H5RcWUThB3fx1fD5fUop	Creep - mint Singles	spotify:track:3H5RcWUThB3fx1fD5fUop
8 2	Save Me Tonight	2019-01-25	170079	1nufXiTCATAejfO9hLHvO8	Save Me Tonight	spotify:track:1nufXiTCATAejfO9hLHvO8
8 3	We Like To Party	2019-05-17	172380	0l3wIS5dlpw0hv5zhxN0pz	We Like To Party	spotify:track:0l3wIS5dlpw0hv5zhxN0pz
8 4	Let's Love	2020-09-11	200645	6lhZLbb0czULpjb2kFryPS	Let's Love	spotify:track:6lhZLbb0czULpjb2kFryPS
8 5	Kids (feat. MKLA)	2020-08-14	184687	3rgTS3933IMWoPiN6CW4qY	Kids (feat. MKLA)	spotify:track:3rgTS3933IMWoPiN6CW4qY

	track_album_release_date	track_duration_ms	track_id	track_name	track_uri	
8 6	No Chains	2018-04-18	22 26 51	2B8OrKU 8suMxTx 2PcuuSIG	No Chains	spotify:track:2B8OrKU8suMxTx2PcuuSIG
8 7	Beast Mode	2019-06-10	16 16 00	4KxqICJNj BNwsmgz LFy3Is	Beast Mode	spotify:track:4KxqICJNjBNwsmgzLFy3Is
8 8	Tumba	2019-07-26	19 62 00	38Li8b8i0 ZNeUxpV HWXX8q	Tumba	spotify:track:38Li8b8i0ZNeUxpVHWXX8q
8 9	Betoven	2019-03-08	17 40 00	0ge3RAp y254QnS 4wPfgoh K	Betoven	spotify:track:0ge3RApy254QnS4wPfgohK
9 0	damn Right Pt. 2	2020-10-09	18 66 66	3sylvqdN RjbFcWkE MRRT1d	damn Right Pt. 2	spotify:track:3sylvqdNRjbFcWkEMRRT1d
9 1	The Squat Song (with MOTi)	2020-06-26	14 71 42	3w11aMZ Lqp0Wx7 viPz0bjc	The Squat Song (with MOTi)	spotify:track:3w11aMZLqp0Wx7viPz0bjc
9 2	I Don't Wanna Know	2020-05-29	18 05 22	0hHcQSd twycNOe c82X2QId	I Don't Wanna Know	spotify:track:0hHcQSdwtwycNOec82X2QId
9 3	Make It To Heaven (with Raye)	2019-11-21	17 23 80	5dS2dvX VPe7AR6 UwBpno mW	Make It To Heaven (with Raye)	spotify:track:5dS2dvXVPe7AR6UwBpnomW
9 4	The Best In The World Pack	2019-06-15	20 54 26	5ry2OE6R ZzPQFDO 85XkgRb	Money In The Grave (Drake ft. Rick Ross)	spotify:track:5ry2OE6R2zPQFDO85XkgRb
9 5	Rain	2020-10-09	17 42 83	4SGCk2N RPEueHe upJildy4	Rain	spotify:track:4SGCk2NRPEueHeupJildy4
9 6	Hollywood's Bleeding	2019-09-06	22 95 73	7CZyCXK G6d5ALe q41sLzbw	Take What You Want	spotify:track:7CZyCXKG6d5ALeq41sLzbw
9 7	Colores	2020-03-19	15 77 33	6zEgnpM 0qYmHL Dnh8WPe jL	Amarillo	

