**Project Title:** Personalized Event Ticket Generator with QR Code

### **Objective:**

Create a Python-based command-line program that generates a personalized event ticket based on user input, formats the output, and saves a QR code representing the ticket data.

### **Features:**

* Input: User’s name, age, and favorite artist
* Process:
  + String formatting and manipulation
  + Ticket code generation
  + Escape character usage for formatting
  + QR code generation and saving as PNG
* Output:
  + Printed ticket details
  + Saved QR code image

### **Technologies Used:**

* Python 3.x
* qrcode library
* PIL (installed via qrcode[pil])
* Regular expressions (re module) for filename sanitization

### **Problem Scenario (Real-time Use Case):**

You’re managing a music festival and need a simple Python script to generate a digital ticket that includes:

* Attendee info
* A unique ticket code
* A QR code for scanning at the gate

### **Initial Code (Before Fix):**

qr\_filename = f"{formatted\_name}\_ticket\_qr.png"  
qr.save(qr\_filename)

### **Error Encountered:**

OSError: [Errno 22] Invalid argument: '...\_ticket\_qr.png'

**Cause:** The filename formatted\_name might include unsafe characters (e.g., &, :, /, \) that are not allowed in filenames, especially on Windows.

### **Solution Applied:**

Sanitize the filename by removing all non-alphanumeric characters using a regular expression:

import re  
safe\_name = re.sub(r'\W+', '', formatted\_name)  
qr\_filename = f"{safe\_name}\_ticket\_qr.png"  
qr.save(qr\_filename)

This ensures the filename is safe and avoids OSError.

### **Final Working Code:**

import qrcode  
import re  
  
# 1. User input  
name = input("Enter your name: ")  
age = int(input("Enter your age: "))  
artist = input("Enter your favorite artist: ")  
  
# 2. Format data  
formatted\_name = name.capitalize()  
reversed\_artist = artist[::-1].upper()  
ticket\_code = formatted\_name[:3].upper() + str(age) + formatted\_name[-2:].upper()  
  
# 3. Message formatting  
message = f"Hello {formatted\_name}, we're thrilled to have you at the festival!\n" \  
 f"Your artist in reverse is: {reversed\_artist}\n" \  
 f"Enjoy your day!\tSee you at Gate {age % 5 + 1}!\n"  
  
# 4. Ticket details for QR  
ticket\_info = f"""  
MUSIC FESTIVAL TICKET  
---------------------  
Name: {formatted\_name}  
Age: {age}  
Favorite Artist: {artist}  
Ticket Code: {ticket\_code}  
"""  
  
# 5. QR Code Generation  
safe\_name = re.sub(r'\W+', '', formatted\_name)  
qr\_filename = f"{safe\_name}\_ticket\_qr.png"  
qr = qrcode.make(ticket\_info)  
qr.save(qr\_filename)  
  
# 6. Output Ticket  
print("\n\U0001F39B MUSIC FESTIVAL TICKET \U0001F39B")  
print("=" \* 27)  
print(f"\nName: {formatted\_name}")  
print(f"Age: {age}")  
print(f"Favorite Artist: {reversed\_artist} (original: \"{artist}\")\n")  
print("Your customized message:\n" + message)  
print(f"Your Ticket Code: {ticket\_code}")  
print(f"\n[\u2705] Your QR Code has been saved as '{qr\_filename}'")  
print("\n" + "=" \* 27)

### **Line-by-Line Explanation:**

* import qrcode, re: Import libraries for QR code creation and regex.
* input(...): Get name, age, and artist from user.
* capitalize(), [::-1], upper(): Format and reverse strings.
* ticket\_code: Use slicing and age to create a unique code.
* \n, \t: Add line breaks and tabs to the message.
* re.sub(r'\W+', '', ...): Remove special characters from name for safe filename.
* qrcode.make(...): Generate QR code with ticket info.
* qr.save(...): Save the QR image.
* print(...): Display all information clearly.

### **QR Code Example:**

Below is a sample QR code image generated by the script:



### **Conclusion:**

This project demonstrates the power of Python basics combined with real-world applications like QR code generation and string formatting. By debugging and fixing a filename error, we made the script production-ready and more robust for any user input.