



(<https://mertmekatronik.com/>)

Python (<https://mertmekatronik.com/yazilim/python>)

.py Uzantılı Dosyanın Uzantisını .exe Ayarlama

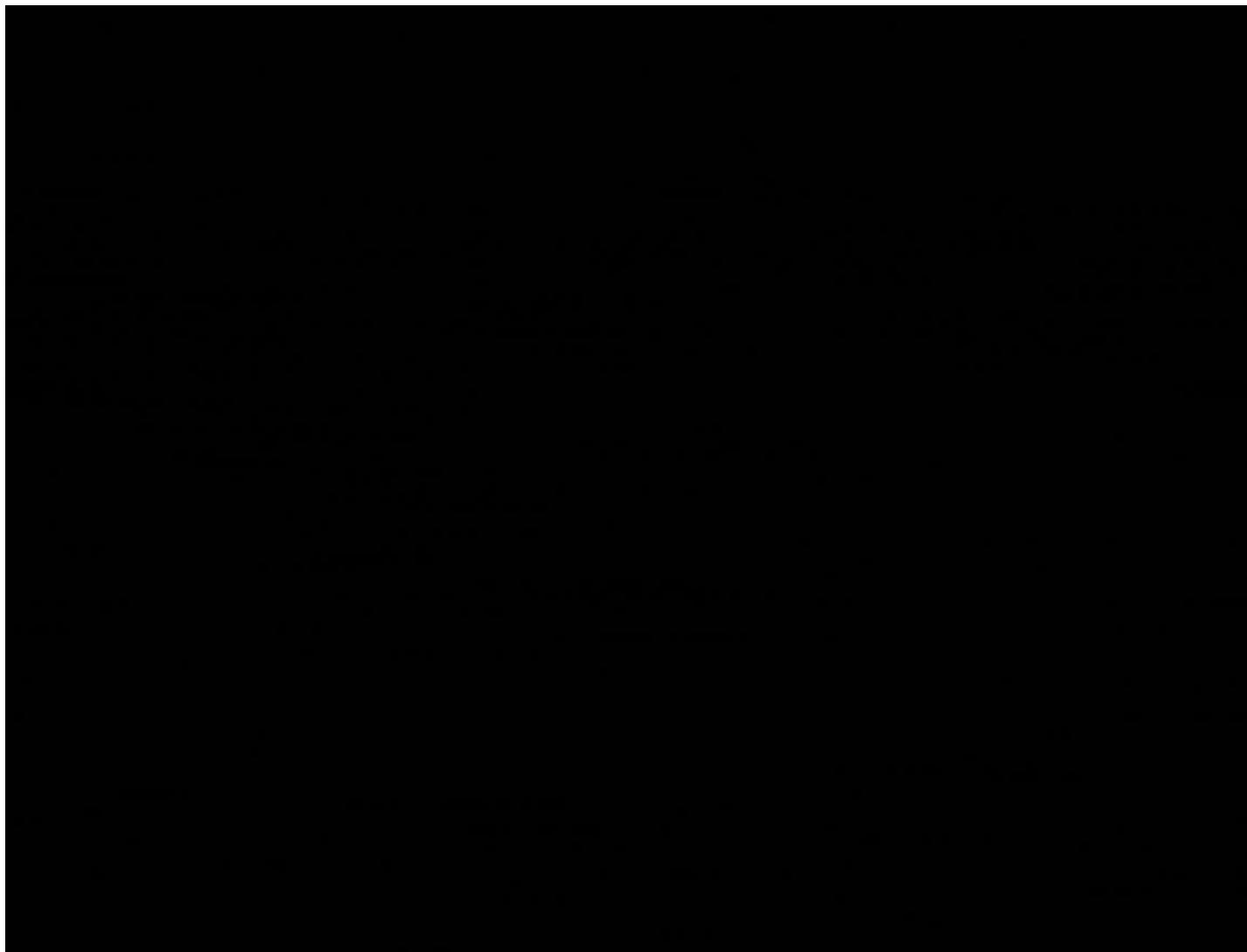
auto-py-to-exe nasıl kullanılır? .py'den .exe'ye



Mücahit Gündüz (<https://mertmekatronik.com/profil/mucahit-gunduz>) Ocak 16, 2021 - 12:38

Güncellenmiş: Nisan 15, 2021 - 11:10

0 577



py to exe

İçerik



- Kullanılacak Python Modülü'nün Kurulması.

≡ Modülü Tanıyalım.

- Modülün Arayüzü (<https://mertmekatronik.com/>)



- Script Location
 - Onefile
 - Console Window
 - Console Based
 - Window Based
 - Icon
 - Additional Files
 - Advanced
 - General Options
 - What to bundle, where to search
 - How to generate
 - Windows specific options
 - Windows Side-by-side Assembly searching options (advanced)
 - Mac OS X specific options
 - Rarely used special options
 - Settings
 - auto-py-to-exe Specific Options
 - Manually Provide Options
 - Configuration
 - Current Command
- Küçük Bir Kod Yazıp Deneyelim (Konsol)
 - Küçük Bir Kod Daha Yazıp Deneyelim (GUI)

Kullanılacak Python Modülü'nün Kurulması

Kullanacağımız Python modülünün ismi "auto-py-to-exe" (<https://pypi.org/project/auto-py-to-exe/>).

Kurmak için terminal açıp

```
C:\ Komut İstemi
Microsoft Windows [Version 10.0.19041.685]
(c) 2020 Microsoft Corporation. Tüm hakları saklıdır.

C:\Users\myygu>pip install auto-py-to-exe
```

veya



```
Komut İstemi
Microsoft Windows [Version 10.0.19041.685]
(c) 2020 Microsoft Corporation. Tüm hakları saklıdır.

C:\Users\myygu>python -m pip install auto-py-to-exe
```

yazmanız gereklidir. Eğer Python bilgisayarınızda yanlış kurulmamışsa modülümüz inecektir. Python kurulumu yanlış ise [buradan](https://www.mertmekatronik.com/python-nasil-kurulur) (<https://www.mertmekatronik.com/python-nasil-kurulur>) Python'nın bilgisayara nasıl kurulacağını öğrenebilirsiniz.

Modülü Tanıyalım

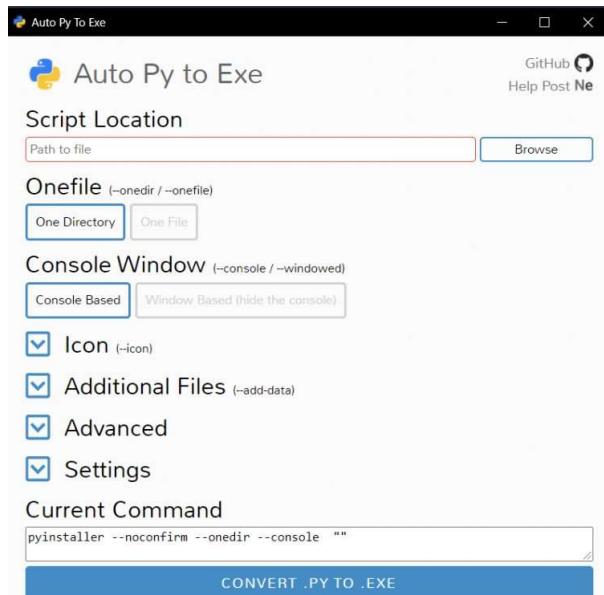
"auto-py-to-exe" modülünü kullanmadığımız sebebi bize bir arayüz sağlama. Arayüzün önemi .exe'ye çevirmek için bilgileri derlerken bize zaman kazandırması. Modülü çalıştırmak için terminalinize

```
Komut İstemi
Microsoft Windows [Version 10.0.19041.685]
(c) 2020 Microsoft Corporation. Tüm hakları saklıdır.

C:\Users\myygu>auto-py-to-exe
```

(<https://resimyukle.xyz/i/aW1BzB>)

yazmanız yeterli olacaktır. Modülü çalıştırduğumuzda karşımıza bir arayüz çıkacaktır.



Modülün Arayüzü Tanıyalım

- Script Location : .py uzantılı dosyamızın dosya yolunu girdiğimiz alan.
- Onefile : Burada karşımıza iki buton çıkıyor;

- One Directory: Gerekli dosyalar .exe dosyasının yanında yer alır.
- One File: Gerekli dosyaların .exe dosyasının içerisinde yer alır(eğer bir GUI uygulaması yapıyorsanız)(<https://mertmekatronik.com/>)
- Console Window: Burada da karşımıza iki buton çıkarıyor;
 - Console Based: .exe dosyasını çalışırken terminalin gözükmesini sağlar.
 - Window Based: .exe dosyasını terminal olmadan çalıştırır(çoğunlukla GUI türü uygulamalarda kullanılır).
- Icon : .exe dosyasının simgesini ayarladığımız alan.
- Additional Files: .exe yaptığımız dosyanın gerekli data dosyalarını belirttiğimiz alan(çoğunlukla One File seçeneği seçilen dosyalarda kullanılır).
- Advanced: Bu alan derleme yaparken uygulamanın genel bilgilerini (uygulama ismi, version bilgisinin bulunduğu dosya vs.) tanımladığımız alan.
 - General Options :
 - --name : Paketlenen uygulamaya ve spesifikasyon dosyasına atanacak ad (varsayılan: ilk komut dosyasının temel adı).
 - --upx-dir : UPX yardımcı programının yolu (varsayılan: yürütme yolunu arayın).
 - --ascii : Unicode kodlama desteğini dahil etmeyin (varsayılan: varsa dahil edilir).
 - --clean : PyInstaller önbelleğini temizleyin ve oluşturmadan önce geçici dosyaları kaldırın.
 - --log-level : Derleme zamanı konsol mesajlarındaki ayrıntı miktarı. TRACE, DEBUG, INFO, WARN, ERROR, CRITICAL (varsayılan: INFO) olabilir.
 - What to bundle, where to search:
 - --add-binary : Yürüttürebilir dosyaya eklenecek ek ikili dosyalar. Daha fazla ayrıntı için Additional Files alanına bakın. Bu alan birden çok kez kullanılabilir.
 - --paths : İthalatları aramak için bir yol (PYTHONPATH kullanmak gibi). Birden fazla yola izin verilir, ":" ile ayrılır veya bu seçeneği birden çok kez kullanın.
 - --hidden-import : Betik kodunda görünmeyen bir içe aktarmayı adlandırın. Bu seçenek birden çok kez kullanılabilir.
 - --additional-hooks-dir : Kancaları aramak için ek bir yol. Bu seçenek birden çok kez kullanılabilir.
 - --runtime-hook : Özel bir çalışma zamanı kanca dosyasının yolu. Bir çalışma zamanı kancası, yürütülebilir dosya ile birlikte paketlenmiş olan ve çalışma zamanı ortamının özel özelliklerini ayarlamak için



herhangi bir başka kod veya modülden önce çalıştırılan koddur. Bu seçenek birden çok kez kullanılabılır.

- --exclude-modules: ~~Yok sayınca istenilen modül veya paket (Python adı, yol adı değil) (sanki bulunamamış gibi).~~ Bu seçenek birden çok kez kullanılabilir.
- --key : Python bayt kodunu şifrelemek için kullanılan anahtar.
- How to generate:
 - --debug : Dondurulmuş bir uygulamada hata ayıklama konusunda yardım sağlayın. Bu bağımsız değişken, aşağıdaki seçeneklerden birkaçını seçmek için birden çok kez sağlanabilir.
 - all : Aşağıdaki seçeneklerin üçü de.
 - imports : içe aktarmalar: temeldeki Python yorumlayıcısına -v seçeneğini belirterek, bir modül her başlatıldığından bir mesaj yazdırmasına ve yüklediği yeri (dosya adı veya yerleşik modül) göstermesine neden olur.
 - bootloader: önyükleyiciye, paketlenmiş uygulamayı başlatırken ve başlatırken ilerleme mesajları vermesini söyleyin. Eksik ithalatlarla ilgili sorunları teşhis etmek için kullanılır.
 - noarchive: dondurulmuş tüm Python kaynak dosyalarını ortaya çıkan yürütülebilir dosyanın içinde bir arşiv olarak depolamak yerine, bunları ortaya çıkan çıktı dizininde dosyalar olarak saklayın.
 - --strip : Yürüttürebilir ve paylaşılan kitaplıklara bir simbol tablosu şeridi uygulayın (Windows için önerilmez).
 - --noupx : Mevcut olsa bile UPX kullanmayın (Windows ve * nix arasında farklı çalışır).
 - --upx-exclude : Upx kullanırken bir ikilinin sıkıştırılmasını önleyin. Bu genellikle, upx sıkıştırma sırasında belirli ikili dosyaları bozarsa kullanılır. FILE, yolsuz ikili dosyanın adıdır. Bu seçenek birden çok kez kullanılabilir.
- Windows specific options:
 - --version-file : .exe'ye bir sürüm kaynağı ekleyin.
 - --manifest : .exe'ye manifest DOSYA veya XML ekleyin.
 - --resource : Bir Windows yürütülebilir dosyasına bir kaynak ekleyin veya güncelleyin.
 - --uac-admin : Bu seçeneğin kullanılması, uygulama yeniden başlatıldığından yükseltme isteyecek bir Manifest oluşturur.
 - --uac-uiaccess : Bu seçeneğin kullanılması, yükseltilmiş bir uygulamanın Uzak Masaüstü ile çalışmasına izin verir.



- Windows Side-by-side Assembly Archiving options (advanced):
 - --win-private-assemblies : Uygulamaya dahil edilen tüm Paylaşılan Montajlar, Özel Montajlar ararken PyInstaller, daha yeni sürümlere yeniden yönlendiren ilkeleri takip etmemeyi tercih edecek ve derlemenin tam sürümlerini paketlemeye çalışacaktır.
 - --win-no-prefer-redirects : Uygulamaya paketlenecek Paylaşilan veya Özel Montajlar ararken PyInstaller, daha yeni sürümlere yeniden yönlendiren ilkeleri takip etmemeyi tercih edecek ve derlemenin tam sürümlerini paketlemeye çalışacaktır.
- Mac OS X specific options:
 - --osx-bundle-identifier : Mac OS X .app paket tanımlayıcısı, kod imzalama amaçları için varsayılan benzersiz program adı olarak kullanılır. Olağan biçim, ters DNS gösterimindeki hiyerarşik bir addır. Örneğin: com.mycompany.department.appname (varsayılan: ilk komut dosyasının temel adı).
- Rarely used special options:
 - --runtime-tmpdir : Nerede kütüphaneler ve destek dosyaları ayıklamak için *onefile* -mode. Bu seçenek verilirse, önyükleyici, çalışma zamanı işletim sistemi tarafından tanımlanan herhangi bir geçici klasör konumunu yok sayar. _MEIxxxxxx Klasöründe dosyalanır burada oluşturulacaktır. Lütfen bu seçeneği yalnızca ne yaptığınızı biliyorsanız kullanın.
 - --bootloader-ignore-signals : Bootloader'a sinyalleri alt sürece iletmek yerine yok olmasını söyleyin. Örneğin bir süpervizör işleminin, çocuğu iki kez sinyal vermekten kaçınmak için hem önyükleyiciyi hem de çocuğu (örneğin bir işlem grubu aracılığıyla) işaret ettiği durumlarda kullanışlıdır.
- Settings:
 - auto-py-to-exe Specific Options:
 - Output Directory : .exe dosyamızın çıkartılacağı dosya yolu girilen alan.
 - Increase Recursion Limit : Bir Özyineleme Hatası meydana gelirse, derinliği 5000'e ayarlamak için bunun etkinleştirildiğinden emin olun.
 - Manually Provide Options :
 - Manual Argument Input : Mevcut komuta manuel olarak eklemek için basit bir giriş.
 - Configuration : Aynı verileri kullanıcı arayüzüne defalarca eklemek yerine, gelişmiş sekmenin altındaki "Konfigürasyon İçe Aktarma(Import Config

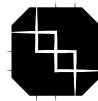


From JSON File) ve Dışa Aktar (Export Config To JSON File)" bölümüne gidip panoya bir JSON dizesi vererek arayüzün mevcut durumunu dışa aktarabilirsiniz veya <https://mertmekatronik.com/> tüm alanları yeniden doldurmak için kullanıcı arayüzüne yeniden aktarılabilir.

- Current Command:
 - CONVERT .PY TO .EXE: Derleme işlemini bitirdikten sonra bu butona basıp .py'den .exe'ye çevirme işlemini yapmaya başlar.
 - CLEAR OUTPUT: .py'den .exe'ye çevirme işlemi bittiğinden sonra karşımıza çıkan butonlardan biri çıkıştır yani Console'u temizlemeye yarar.
 - OPEN OUTPUT FOLDER: Settings alanında ki Output Directory alanında yazan dosya ya sizi hızlı bir şekilde götürür.

Küçük Bir Kod Yazıp Deneyelim (Konsol)





```
import sys
import time

class Register:
    def __init__(self):
        self.username=input("Kullanıcı adı:")
        while True:
            self.password=input("Şifre:")
            self.againpassword=input("Şifre tekrar:")
            if self.password==self.againpassword:
                self.save()
                break
            else:
                print("Şifreler uyumsuz tekrar deneyin")

    def save(self):
        user_infos=[f"Kullanıcı adı :{self.username}",f"Şifre :{self.password}"]
        with open(f"{self.username}_bilgileri.txt","w",encoding="utf-8") as info_file:
            for user_info in user_infos:
                print(user_info,file=info_file)

class Login:
    def __init__(self):
        self.username=input("Kullanıcı adı :")
        self.password=input("Şifre :")
        control=self.controlinfo()
        if control:
            print("Giriş başarılı hoşgeldin")
            print("10 saniye sonra otomatik olarak kapanacaktır!!!")
            time.sleep(10)
            sys.exit()
        else:
            print("Giriş başarısız")

    def controlinfo(self):
        try:

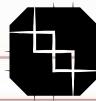
            with open(f"{self.username}_bilgileri.txt","r",encoding="utf-8") as info_file:
                user_infos=info_file.readlines()
            if user_infos[0]==f"Kullanıcı adı :{self.username}\n" and user_infos[1] ==f"Şifre :{self.password}\n":
                return True
        except:
            print("Giriş başarısız")
            Login()

while True:
    print("Hoşgeldiniz Kayıt için 'K' yi Giriş için 'G' yi girin")
    selection=input(":")
    if selection.upper().strip() == "K":
        Register()
    elif selection.upper().strip() == "G":
        Login()
```

Küçük bir kodumuz var. Bu kodun uzantisini .py'den .exe'ye çevirelim.

1.Adım: .py dosyamızı seçme.

Script Location



Path to file

Browse



(<https://mertmekatronik.com/>)

Browse tuşuna basıp .py dosyamızı seçelim.

Script Location

C:/Users/myygu/OneDrive/Masaüstü/py'den exe'ye/makale.py

Browse

2.Adım: Gerekli dosyalar yanında mı(One Directory) olsun yoksa içinde mi(One File) olsun?

Onefile (`--onedir / --onefile`)

One Directory

One File

Bu örnekte "One Directory" butonunu seçiyoruz çünkü gerekli dosyaları .exe dosyasının yanında koyucaz özür örnekte "One File" butonunu kullanarak yapıcız.

3.Adım: Konsol tabanlı(Console Based) mı Pencere tabanlı(Window Based) mı?

Console Window (`--console / --windowed`)

Console Based

Window Based (hide the console)

Bu örnekte "Console Based" butonunu seçiyoruz çünkü kodumuz bir GUI modülü kullanılmadığı için konsolda çalışacak.

4.Adım: Simge(Icon) seçme.



Icon (`--icon`)

.ico file

Browse

Eğer .exe yapıcığınız programın simgesi olmasını istiyorsanız "Browse" tuşuna basarak bilgisayarınızdan simgenizi seçebilirsiniz.

5.Adım: Gerekli dosyaları path e ekleme.



Additional Files (`--add-data`)

Add Files

Add Folder

Add Blank

If you want to put files in the root directory, put a period(.) in the destination.

Bu bölümü şu an doldurmayacağımız çünkü örnek olarak hazırladığımız programın dışardan aldığı bir dosya yok.

6.Adım: Advanced bölümü.



Bu bölümde sadece "--name" kütüğünü doldurmanız yeterli olur. İsterseniz onu da doldurmaya da bilirsiniz. .dül .programa .py dosyasının ismini vericektir. (<https://mertmekatronik.com/>)

7.Adım: Settings bölümü.

auto-py-to-exe Specific Options

Output Directory C:\Users\myygu\output Browse

Increase Recursion Limit Enable

Manually Provide Options

Manual Argument Input ARGUMENTS

Configuration

Import Config From JSON File Export Config To JSON File

Bu bölümde bir değişiklik yapmanıza gerek yok eğer isterseniz .exe nizi istediğiniz dosya içine koymak için dosyanın yolunu "Browse" butonuna tıklayarak dosya konumunu seçebilirsiniz.

8.Adım: Current Command bölümü.

Current Command

```
pyinstaller --noconfirm --onedir --console "C:/Users/myygu/OneDrive/Masaüstü/py'den exe'ye/makale.py"
```

CONVERT .PY TO .EXE

Bu bölüm işlemimizin biteceği bölüm. "CONVERT .PY TO .EXE" butonuna basarak çevirmesini bekleyelim.

9.Adım: .exe dosyamızı bulma.

CLEAR OUTPUT OPEN OUTPUT FOLDER

Bu bölümde "OPEN OUTPUT FOLDER" butonuna basarak .exe programımızın yanına gidebiliriz.

10.Adım: Kontrol.

.exe programımızın bulunduğu klasöre gidelim ve kontrol edelim.

Mücahit Gündüz > output > Konsol Kayıt >

Ad	Tarih	Tür	Boyut
Include	11.01.2021 08:43	Dosya klasörü	
_asyncio	4.12.2020 23:52	Python Extension ...	56 KB
_bz2	4.12.2020 23:52	Python Extension ...	77 KB
_ctypes	4.12.2020 23:52	Python Extension ...	114 KB
_decimal	4.12.2020 23:52	Python Extension ...	223 KB
_hashlib	4.12.2020 23:52	Python Extension ...	38 KB
_lzma	4.12.2020 23:52	Python Extension ...	155 KB
_multiprocessing	4.12.2020 23:52	Python Extension ...	26 KB
_overlapped	4.12.2020 23:52	Python Extension ...	39 KB
_queue	4.12.2020 23:52	Python Extension ...	25 KB
_socket	4.12.2020 23:52	Python Extension ...	68 KB
_ssl	4.12.2020 23:52	Python Extension ...	140 KB
base_library	11.01.2021 08:43	WinRAR ZIP archive	769 KB
Konsol Kayıt	11.01.2021 08:43	Uygulama	1.907 KB
Konsol Kayıt.exe.manifest	11.01.2021 08:43	MANIFEST Dosyası	2 KB
libcrypto-1_1.dll	4.12.2020 23:52	Uygulama uzantısı	2.177 KB
libffi-7.dll	4.12.2020 23:52	Uygulama uzantısı	29 KB
libssl-1_1.dll	4.12.2020 23:52	Uygulama uzantısı	526 KB
pyexpat	4.12.2020 23:52	Python Extension ...	164 KB
python38.dll	4.12.2020 23:52	Uygulama uzantısı	3.957 KB
select	4.12.2020 23:52	Python Extension ...	24 KB
unicodedata	4.12.2020 23:52	Python Extension ...	1.066 KB
VCRUNTIME140.dll	4.12.2020 23:52	Uygulama uzantısı	82 KB

Konsol Kayıt Özellikleri

Genel Uyumluluk Güvenlik Ayrintılar Önceki Sürümler

Konsol Kayıt

Dosya türü: **Uygulama (.exe)**

Açıklama: Konsol Kayıt

Konum: C:\Users\myygu\output\Konsol Kayıt

Boyut: 1,86 MB (1.952.391 bayt)

Diskteki boyut: 1,86 MB (1.953.792 bayt)

Oluşturma: 11 Ocak 2021 Pazartesi, 08:43:15

Değiştirme: 11 Ocak 2021 Pazartesi, 08:43:15

Erişim: 11 Ocak 2021 Bugün, 10 dakika önce

Öznitelikler: Salt okunur Gizli Gelişmiş...

Tamam İptal Uygula

gördüğünüz gibi .py'den .exe'ye çevirme işlemimiz başarıyla tamamladık.

Küçük Bir Kod daha Yazıp Deneyelim (GUI)



```
from PyQt5.QtWidgets import *
from PyQt5.QtGui import *
from PyQt5.QtCore import * (https://mertmekatronik.com/)

import sys
import time
import os

def resource_path(relative_path):
    base_path = getattr(sys, '_MEIPASS', os.path.dirname(os.path.abspath(__file__)))
    return os.path.join(base_path, relative_path)

class Program(QWidget):
    def __init__(self, parent=None):
        super().__init__(parent)
        self.title="Kayıt Programı"
        self.left=800
        self.right=300
        self.width=500
        self.height=300
        self.initUI()
    def initUI(self):
        self.setWindowTitle(self.title)
        self.setGeometry(self.left, self.right, self.width, self.height)
        self.setStyleSheet(open(resource_path("makaleGUI.css"), "r").read())
        self.setWindowIcon(QIcon(resource_path("icon.ico")))
        self.showWindow()

        mainLayout = QVBoxLayout()
        mainLayout.addWidget(self.GroupBox)
        mainLayout.addWidget(self.GroupBox2)
        self.setLayout(mainLayout)
        self.backend()

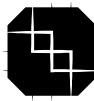
    def window(self):
        #=====LOGIN=====
        self.login_username=QLineEdit()
        self.login_username.setMinimumWidth(300)
        self.login_username.setMinimumHeight(35)
        self.login_username.setPlaceholderText("Kullanıcı adı")

        self.login_password=QLineEdit()
        self.login_password.setMinimumWidth(300)
        self.login_password.setMinimumHeight(35)
        self.login_password.setPlaceholderText("Şifre")
        self.login_password.setEchoMode(QLineEdit.Password)

        self.login_login=QPushButton("Giriş Yap")
        self.login_login.setMaximumWidth(100)
        self.login_login.setMaximumHeight(35)
        self.login_login.clicked.connect(lambda:self.backend("Login"))
        self.login_login.setCursor(QCursor(Qt.PointingHandCursor))

        self.login_register=QPushButton("Kayıt Ol")
        self.login_register.setMaximumWidth(100)
        self.login_register.setMaximumHeight(35)
        self.login_register.clicked.connect(lambda:self.backend("Register"))
        self.login_register.setCursor(QCursor(Qt.PointingHandCursor))
        #=====REGISTER=====

        self.username=QLineEdit()
        self.username.setMinimumWidth(300)
        self.username.setMinimumHeight(35)
        self.username.setPlaceholderText("Kullanıcı adı")
```



```
self.password=QLineEdit()
self.password.setMinimumWidth(300)
self.password.setMinimumHeight(35)
self.password.setPlaceholderText("Şifre")
self.password.setEchoMode(QLineEdit.Password)

self.againpassword=QLineEdit()
self.againpassword.setMinimumWidth(300)
self.againpassword.setMinimumHeight(35)
self.againpassword.setPlaceholderText("Şifre Tekrar")
self.againpassword.setEchoMode(QLineEdit.Password)

self.login=QPushButton("Giriş Yap")
self.login.setMaximumWidth(100)
self.login.setMaximumHeight(35)
self.login.clicked.connect(lambda:self.backend("Login"))
self.login.setCursor(QCursor(Qt.PointingHandCursor))

self.register=QPushButton("Kayıt Ol")
self.register.setMaximumWidth(100)
self.register.setMaximumHeight(35)
self.register.clicked.connect(lambda:self.backend("Register"))
self.register.setCursor(QCursor(Qt.PointingHandCursor))

def showWindow(self):
    self.window()
    self.GroupBox = QGroupBox()
    layout = QGridLayout()
    layout.addWidget(self.login_username,0,0)
    layout.addWidget(self.login_password,1,0)
    layout.addWidget(self.login_login,2,0,alignment=Qt.AlignRight)
    layout.addWidget(self.login_register,2,1)

    self.GroupBox.setLayout(layout)

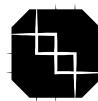
    self.GroupBox2 = QGroupBox()
    layout = QGridLayout()
    layout.addWidget(self.username,0,0)
    layout.addWidget(self.password,1,0)
    layout.addWidget(self.againpassword,2,0)
    layout.addWidget(self.login,3,0,alignment=Qt.AlignRight)
    layout.addWidget(self.register,3,1)

    self.GroupBox2.setLayout(layout)

def backend(self,mode="Login"):
    if mode=="Register":
        if len(self.username.text().strip())==0 and len(self.password.text().strip())==0 and
            self.login_password.clear()
            self.login_username.clear()
            self.GroupBox.hide()
            self.GroupBox2.show()
    else:
        if self.password.text()==self.againpassword.text() and len(self.username.text()):
            user_infos=[f"Kullanıcı adı :{self.username.text()}",f"Şifre :{self.password.text()}"
            with open(f"{self.username.text()}_bilgileri.txt","w",encoding="utf-8") as info_file:
                for user_info in user_infos:
                    print(user_info,file=info_file)
            infomessege=QMessageBox.information(self, "Sistem", "Kayıt Başarıyla Tamamlandı")
```

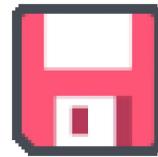
```
else:  
    errormessege=QMessageBox.warning(self, "Hata", "Şifre Uyumsuzluğu !!!")  
    self.password.clear()  
    self.againpassword.clear()  
\(https://mertmekatronik.com/\)  
  
else:  
    if len(self.login_username.text().strip())==0 and len(self.login_password.text().strip())==0:  
        self.password.clear()  
        self.againpassword.clear()  
        self.username.clear()  
        self.GroupBox2.hide()  
        self.GroupBox.show()  
    else:  
        self.control=True  
        try:  
            with open(f"{self.login_username.text().strip()}_bilgileri.txt","r",encoding="utf-8") as info_file:  
                user_infos=info_file.readlines()  
            if user_infos[0]==f"Kullanıcı adı :{self.login_username.text().strip()}\n" and user_infos[1]==f"Şifre :{self.login_password.text().strip()}\n":  
                self.control= True  
            else:  
                self.control=False  
        except:  
            errormessege=QMessageBox.warning(self, "Hata", "Kullanıcı adı veya Şifre yanlış")  
            self.control = False  
    if self.control:  
        infomessege=QMessageBox.information(self, "Sistem", "Giriş başarılı hoş geldiniz")  
        time.sleep(10)  
        sys.exit()  
  
if __name__ == '__main__':  
    app = QApplication(sys.argv)  
    win = Program()  
    win.show()  
    sys.exit(app.exec_())
```

PyQt5 modülü ile hazırlanmış, küçük bir kodumuz var.



```
QWidget{
    background-color:#53617A;
    color: White;
    font-size: 15px;
}
QPushButton{
    background-color: White;
    color:#53617A;
    font-family: serif;
    font-weight: bold;
}
QLineEdit{
    border: 3px solid white;
    border-left: 10px;
    border-right: 10px;
    border-top: 10px;
}
QGroupBox{
    border:0px solid white;
}
```

Bu css kodları .py dosyamıza kullandığımız css kodları.



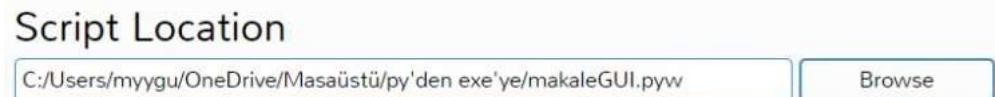
Bu simge .py dosyamızda kullandığımız simge. (.ico uzantılı hali kullanılmıştır.)

Adım adım neler yapmamıza gerektiğini görelim.

1.Adım: .py dosyamızı seçme.

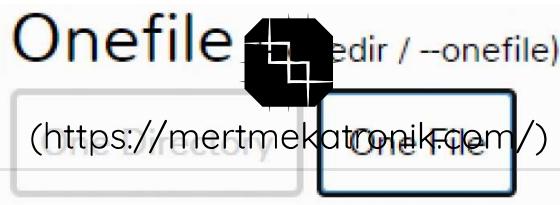


Browse tuşuna basıp .py dosyamızı seçelim.



.py dosyamızın uzantısı .py yerine .pyw yazmakta bunun sebebi .exe' ye çevirmeden önce konsolu gizlemiş olmamız. 3.Adımda bahsedeceğiz bundan.

2.Adım: Gerekli dosyalar yanında mı(One Directory) olsun yoksa içinde mi(One File) olsun?



Bu bölümde "One File" butonunu seçicez çünkü gerekli dosyaların içinde olmasını istiyoruz.

3.Adım: Konsol tabanlı(Console Based) mı Pencere tabanlı(Window Based) mı?

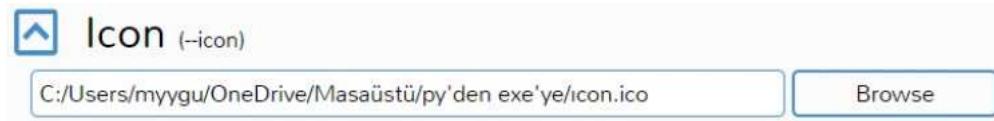


Bu bölümde "Concole Based" butonunu seçmemizin sebebi "Window Based" butonu seçilerek .exe'ye çevrilen programların hata verip kullanılamaması. Biz de bu soruna çözüm olarak .py uzantisını .pyw yapıyoruz böylelikle hata vermeden konsolu gizlemiş oluyoruz.

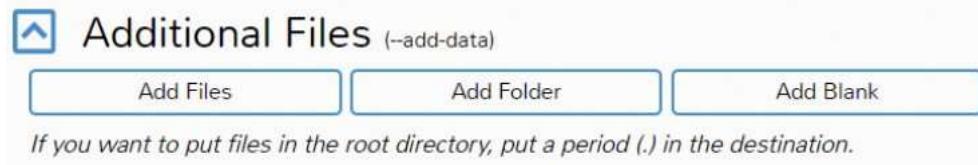
4.Adım: Simge(Icon) seçme.



Bu bölümde elimizde simge olduğu için ekleyeceğiz isterseniz eklemeyedebilirsiniz.



5.Adım: Gerekli dosyaları path e ekleme.



Bu bölümde, 2.Adımda "One File" butonunu seçtiğimiz için gerekli dosyaları eklememiz gerekiyor. Onun için ayarladığınız dosya yoluna göre "Add Files" veya "Add Folder" butonlarına basarak gerekli dosyaları seçebilirsiniz.

Additional Files

Add Files Add Folder Add Blank

C:/Users/myygu/OneDrive/Masaüstü/py'de (https://mertmekatronik.com/)

C:/Users/myygu/OneDrive/Masaüstü/py'de .

6.Adım: Advanced bölümü.

Bu bölümde önceki örnekteki gibi sadece "--name" bölümünü doldurmanız yeterli olur. İsterseniz onu da doldurmaya da bilirsiniz modül programa .py dosyasının ismini vericektir.

7.Adım: Settings bölümü.

Settings

auto-py-to-exe Specific Options

Output Directory C:\Users\myygu\output

Increase Recursion Limit

Manually Provide Options

Manual Argument Input ARGUMENTS

Configuration

Bu bölümde önceki örnek gibi bir değişiklik yapmanıza gerek yok eğer isterseniz .exe nizi istediğiniz dosya içine koymak için dosyanın yolunu "Browse" butonuna tıklayarak dosya konumunu seçebilirsiniz.

8.Adım: Current Command bölümü.

Current Command

```
pyinstaller --noconfirm --onefile --console --icon
"C:/Users/myygu/OneDrive/Masaüstü/py'de\exe'ye/icon.ico" --name "MakaleGUI"
```

Bu bölümde derleme işlemini bitirdiğimize emin olduktan sonra "CONVERT .PY TO .EXE" butonuna basarak .exe'ye çevirme işlemini başlatalım.

9.Adım: .exe dosyamızı bulma.

Bu bölümde "OPEN OUTPUT FOLDER" butonuna basarak .exe programımızın yanına gidebiliriz.

10.Adım: Kontrol.

The screenshot shows a file management interface with a table header: Ad (Name), Tarih (Date), Boyut (Size), and Etiketler (Tags). A file named 'MakaleGUI' is listed with a date of 12.01.2021 12:03, a size of 29.681 KB, and a red box highlighting the 'Uygulama' (Application) tag under Etiketler.

(<https://mertmekatronik.com/>)

The properties dialog has tabs for Genel (General), Uyumluluk (Compatibility), Güvenlik (Security), Ayrintılar (Details), and Önceki Sürümeler (Previous Versions). The 'Genel' tab is selected. It shows the file name 'MakaleGUI', file type 'Uygulama (.exe)' (highlighted with a red box), and other details like location (C:\Users\myygu\output), size (28.9 MB), and creation date (12 Ocak 2021 Salı, 12:03:57).

Gördüğünüz gibi .py'den .exe'ye dönüştürme işlemimiz başarıyla tamamladık.

Daha fazla bilgi için tıklayın → (<https://pypi.org/project/auto-py-to-exe/>)

Etiketler:

#auto-py-to-exe (<https://mertmekatronik.com/etiket/auto-py-to-exe>)

#Python (<https://mertmekatronik.com/etiket/python>)

◀ ÖNCEKİ MAKALE

(<https://mertmekatronik.com/roket-ve-motorlari-676>)

Roket Ve Motor Teknolojilerine Genel Bakış (<https://mertmekatronik.com/roket-ve-motorlari-676>)

SIRADAKİ MAKALE ▶

(<https://mertmekatronik.com/kullanici-deneyimi-ux>)

Kullanıcı Deneyimi(User Experience - UX) Nedir? Temelleri nelerdir? Kullanıcı Deneyimi... 

(<https://mertmekatronik.com/kullanici-deneyimi-ux>)

NE DÜŞÜNÜYORSUNUZ?



(16)



(0)



(6)



(1)

Beğendim**Beğenmedim****Sevdim****Eğlenceli**

(0)



(0)



(0)

Sınırlendim**Üzüldüm****Şaşırdım**

(<https://mertmekatronik.com/profil/muahit-gunduz>)

Mücahit Gündüz (<https://mertmekatronik.com/profil/muahit-gunduz>)

Yazılım alanında gelişmeye çalışan bir öğrenciyim Python programlama dilinde sahip olduğum bilgileri sizlerle paylaşıyorum.

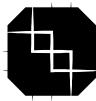
(<https://twitter.com/myggunduz>)

(<https://www.instagram.com/myggunduz/>)

(<https://www.linkedin.com/in/myggunduz/>)

İLGİLİ GÖNDERİLER





(<https://mertmekatronik.com/>)

(<https://mertmekatronik.com/pythonda-listeler>)

Python'da Listelere Ne Kadar Hakimsin? 10 Soruda Listeler

(<https://mertmekatronik.com/pythonda-listeler>)

EmreOzkul (<https://mertmekatronik.com/profil/emreozkul>) Eylül 26, 2020 0 456



(<https://mertmekatronik.com/pythonda-dosya-islemleri>)

Python'da Dosya İşlemleri (<https://mertmekatronik.com/pythonda-dosya-islemleri>)

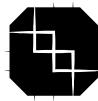
Basakyalciner (<https://mertmekatronik.com/profil/basakyalciner>) Aralik 25, 2020 0 407



(<https://mertmekatronik.com/tkinter-tutorial-7-add-text-edit>)

Tkinter Eğitimi #7-Metin Düzenleme Aracı Eklemek (<https://mertmekatronik.com/tkinter-tutorial-7-add-text-edit>)

Mücahit Gündüz (<https://mertmekatronik.com/profil/mucahit-gunduz>) Haziran 9, 2021 0 176



(<https://mertmekatronik.com/>)

(<https://mertmekatronik.com/pillow-nedir>)

Pillow Modülü Nasıl Kullanılır? (<https://mertmekatronik.com/pillow-nedir>)

Janberk (<https://mertmekatronik.com/profil/janberk>) Temmuz 12, 2021 0 243



(<https://mertmekatronik.com/django-bir-degiskeni-tum-sitede-kullanma-dinamik-footer-yapimi>)



Django Bir Değişkeni Tüm Sitede Kullanma Dinamik Footer (<https://mertmekatronik.com/django-bir-degiskeni-tum-sitede-kullanma-dinamik-footer-yapimi>)

saidtaylann (<https://mertmekatronik.com/profil/saidtaylann>) Temmuz 23, 2020 0 345

(<https://mertmekatronik.com/pyqt5-sinavi>)

PyQt5 Sınavı : Python ile Uygulama Yapımı (<https://mertmekatronik.com/pyqt5-sinavi>)

Ömer F. Yücel (<https://mertmekatronik.com/profil/iostreamCpp>) Ekim 27, 2020 1 919

YORUMLAR

Ad

Ad

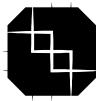
E-Posta

E-Posta

Yorum



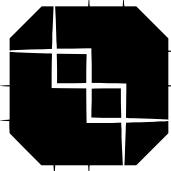
Yorum

[\(https://mertmekatronik.com/\)](https://mertmekatronik.com/)

Ben robot değilim

reCAPTCHA
Gizlilik - Şartlar

Yorum Gönder

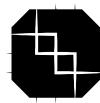


Mert Mekatronik, fikren amacı; kullanıcılarına ve müşterilerine eğitimleri kaliteli ve çeşitli hale getirmek. Türkiye'nin en kapsamlı yazılım, mühendislik, teknoloji platformu, eğitimcisi ve danışmanlık şirketi olmaktadır.

RASTGELE GÖNDERİ

[Arduino ve Coolterm ile Veri Kaydetme](#)[\(https://mertmekatronik.com/saving-data-arduino-coolterm\)](https://mertmekatronik.com/saving-data-arduino-coolterm)[\(https://mertmekatronik.com/saving-data-arduino-coolterm\)](https://mertmekatronik.com/saving-data-arduino-coolterm)[JWT\(JSON Web Token\) Nedir? \(https://mertmekatronik.com/jwt-nedir\)](#)[\(https://mertmekatronik.com/jwt-nedir\)](https://mertmekatronik.com/jwt-nedir)[If - Else Yapısı | C++ \(https://mertmekatronik.com/if-else-yapisi-c++\)](#)

[https://mertmekatronik.com/if-
else-yapisi-c](https://mertmekatronik.com/if-else-yapisi-c) (<https://mertmekatronik.com/>)



SOSYAL MEDYA



[Twitter](#) [Instagram](#) (<https://www.mertmekatronik.com/rss-atom.xml>)

(<https://www.mertmekatronik.com/>)

Hemen Abone Ol ve İçeriklerimizi Kaçırm别!

E-posta adresi

Abone Ol

Mert Mekatronik © Copyright 2021 - Her Hakkı Saklıdır

Şartlar & Koşullar (<https://mertmekatronik.com/sartlar-kosullar>)