

Inizia a programmare o [genera](#) codice con l'IA.

## ✓ ITER & Fusion Energy: Un laboratorio interdisciplinare tra Fisica e Data Science

L'attività didattica presentata in questo documento nasce come risposta concreta e approfondimento scientifico della visita svolta al sito ITER di Caldarache (Francia) tra marzo e aprile 2025. ITER (International Thermonuclear Experimental Reactor) è attualmente il più grande progetto internazionale per dimostrare la fattibilità scientifica e tecnologica dell'energia da fusione nucleare: la stessa reazione che alimenta il Sole e le stelle.

Il 10 aprile 2025, ITER ha raggiunto un risultato tecnico e scientifico straordinario: il **modulo settoriale #7**, una struttura alta quanto un edificio di cinque piani e pesante quanto quattro Jumbo Jet a pieno carico (1.350 tonnellate), è stato calato con successo all'interno del pozzo del Tokamak, il cuore pulsante del reattore. L'operazione, descritta dagli esperti come un "record di performance", è avvenuta con tre settimane di anticipo rispetto al cronoprogramma originale, dimostrando eccezionali capacità ingegneristiche, di pianificazione e collaborazione internazionale.

Questa straordinaria operazione è stata possibile grazie al lavoro coordinato e preciso di oltre 150 persone, tra ingegneri, tecnici, squadre specializzate e operatori di gru, provenienti da diversi paesi e realtà aziendali internazionali. Un risultato che testimonia non solo la complessità tecnica del progetto ITER, ma anche il valore della collaborazione internazionale nella ricerca di soluzioni energetiche sostenibili.

Alla luce di queste importanti informazioni, l'attività didattica che segue intende far vivere agli studenti un'esperienza reale e coinvolgente, utilizzando i dati relativi al funzionamento del reattore ITER come caso studio. Gli studenti possono essere guidati attraverso metodologie didattiche data-driven (Inquiry-Based Learning, Project-Based Learning, Service Learning) al fine di analizzare e visualizzare dati scientifici simulati, rafforzando competenze trasversali e tecniche, secondo le indicazioni del MOOC nazionale "Sensori, grafici e scoperte: giocare ed esplorare con i dati a scuola" (ID percorso: 257576, ScuolaFutura, Ministero dell'Istruzione e del Merito).

### Risorse utili per approfondire ulteriormente:

- **Sito ufficiale ITER:** [iter.org](https://www.iter.org)
- **Video ufficiale "Take-off and landing of Sector Module #7":** [YouTube ITER](#)
- **Blog Ecomentality degli studenti del liceo Calini di Brescia su STEM, innovazione e sostenibilità:** [Ecomentality.it](https://ecomentality.it)

La presente attività didattica è un ponte tra ricerca scientifica, formazione STEM e consapevolezza energetica e ambientale, con particolare attenzione agli obiettivi globali di sostenibilità fissati dall'Agenda 2030 delle Nazioni Unite.

## Data Fusion – ITER e Data Science

### Simulazione didattica della produzione energetica del reattore ITER

#### Nota metodologica:

Questi dati sono simulati esclusivamente a scopo didattico e rappresentano una plausibile evoluzione della potenza prodotta dal reattore ITER.

La simulazione si basa sugli obiettivi reali del progetto ITER, che prevede nella fase operativa di raggiungere circa 500 MW di potenza generata e temperature del plasma nell'ordine di 100-150 milioni di °C.

```
# -----
# Data Fusion - ITER e Data Science
# Laboratorio didattico: Simulazione energetica ITER
# (Notebook didattico realizzato da Veronica Cavicchi)
# -----

# Importo librerie Python per analisi dati e grafici:
import pandas as pd # gestione dati e tabelle
import matplotlib.pyplot as plt # visualizzazione grafici
import numpy as np # calcoli numerici
from scipy.stats import linregress # per regressione lineare

# -----
# PARTE 1: Dataset ITER (7 dati introduttivi)
# -----

# Creo dataset introduttivo ITER con dati simulati realistici
dati_iter = pd.DataFrame({
    "Giorno": [1, 2, 3, 4, 5, 6, 7],
    "Temperatura Plasma (milioni di °C)": [100, 105, 110, 107, 115, 120, 118],
    "Energia Prodotta (MW)": [400, 420, 450, 430, 470, 490, 480]
})

# Calcolo efficienza energetica (Energia Prodotta / Temperatura Plasma)
dati_iter["Efficienza (MW per milione °C)"] = (
    dati_iter["Energia Prodotta (MW)"] / dati_iter["Temperatura Plasma (milioni di °C)"]
```

```

)

# Tabella dati (7 valori)
print("\nTabella Dati ITER (7 dati):")
print(dati_iter)

# Grafico introduttivo doppio asse (Energia ed Efficienza)
fig, ax1 = plt.subplots(figsize=(10,6))

# Energia prodotta (asse sinistro)
ax1.set_title("Simulazione ITER: Energia prodotta ed Efficienza (introduzione)", fontsize=16)
ax1.set_xlabel("Giorno di esperimento", fontsize=14)
ax1.set_ylabel("Energia Prodotta (MW)", color='blue', fontsize=14)
ax1.plot(dati_iter["Giorno"], dati_iter["Energia Prodotta (MW)"], marker='o', color='blue', linewidth=2)
ax1.tick_params(axis='y', labelcolor='blue')
ax1.grid(True)

# Efficienza energetica (asse destro)
ax2 = ax1.twinx()
ax2.set_ylabel("Efficienza (MW per milione °C)", color='red', fontsize=14)
ax2.plot(dati_iter["Giorno"], dati_iter["Efficienza (MW per milione °C)"], marker='s', linestyle='--', color='red', linewidth=2)
ax2.tick_params(axis='y', labelcolor='red')

plt.tight_layout()
plt.show()

# Formula efficienza energetica per calcolo manuale
print("\nFormula Efficienza Energetica (calcolo manuale):")
print("Efficienza = Energia Prodotta (MW) / Temperatura Plasma (milioni di °C)")

# -----
# PARTE 2: Approfondimento avanzato (100 dati realistici)
# -----

# Genero dataset avanzato ITER (100 dati simulati realistici)
np.random.seed(42) # riproducibilità risultati
giorni_100 = np.arange(1, 101)
temp_plasma_100 = np.random.normal(125, 10, 100)
energia_prodotta_100 = temp_plasma_100 * 4 + np.random.normal(0, 20, 100)

df_iter_100 = pd.DataFrame({
    "Giorno": giorni_100,
    "Temperatura Plasma (milioni di °C)": temp_plasma_100,
    "Energia Prodotta (MW)": energia_prodotta_100
})

# Calcolo efficienza energetica avanzata (100 dati)
df_iter_100["Efficienza (MW per milione °C)"] = (
    df_iter_100["Energia Prodotta (MW)"] / df_iter_100["Temperatura Plasma (milioni di °C)"]
)

# Visualizzo prime 10 righe della tabella da 100 dati
print("\nTabella avanzata ITER (prime 10 righe su 100 dati):")
print(df_iter_100.head(10))

# Analisi statistica completa dei 100 dati:
media_temp = df_iter_100["Temperatura Plasma (milioni di °C)"].mean()
dev_std_temp = df_iter_100["Temperatura Plasma (milioni di °C)"].std()
media_energia = df_iter_100["Energia Prodotta (MW)"].mean()
dev_std_energia = df_iter_100["Energia Prodotta (MW)"].std()
correlazione = df_iter_100["Temperatura Plasma (milioni di °C)"].corr(df_iter_100["Energia Prodotta (MW)"])

# Regressione lineare (Energia vs Temperatura Plasma)
reg = linregress(df_iter_100["Temperatura Plasma (milioni di °C)"], df_iter_100["Energia Prodotta (MW)"])

# Grafico avanzato completo (100 dati e regressione lineare)
plt.figure(figsize=(12,7))
plt.scatter(df_iter_100["Temperatura Plasma (milioni di °C)"], df_iter_100["Energia Prodotta (MW)"],
            color='green', label='Dati ITER (100 simulazioni)', alpha=0.6)

# Retta regressione lineare
x_reg = np.linspace(temp_plasma_100.min(), temp_plasma_100.max(), 100)
y_reg = reg.intercept + reg.slope * x_reg
plt.plot(x_reg, y_reg, color='purple', linewidth=3, label='Regressione lineare')

plt.title("Analisi avanzata ITER: Energia vs Temperatura Plasma", fontsize=16)
plt.xlabel("Temperatura Plasma (milioni di °C)", fontsize=14)
plt.ylabel("Energia Prodotta (MW)", fontsize=14)
plt.grid(True)
plt.legend(fontsize=12)

# Annotazioni statistiche nel grafico
plt.annotate(f"Correlazione = {correlazione:.2f}",

```

```
plt.annotate('r correlazione = {correlazione:.2f}\n'
            f"Media Temp = {media_temp:.2f} °C, Dev. Std. = {dev_std_temp:.2f}\n"
            f"Media Energia = {media_energia:.2f} MW, Dev. Std. = {dev_std_energia:.2f}",
            xy=(0.05, 0.7), xycoords='axes fraction',
            bbox=dict(boxstyle="round,pad=0.5", fc="yellow", alpha=0.4), fontsize=12)

plt.tight_layout()
plt.show()
```



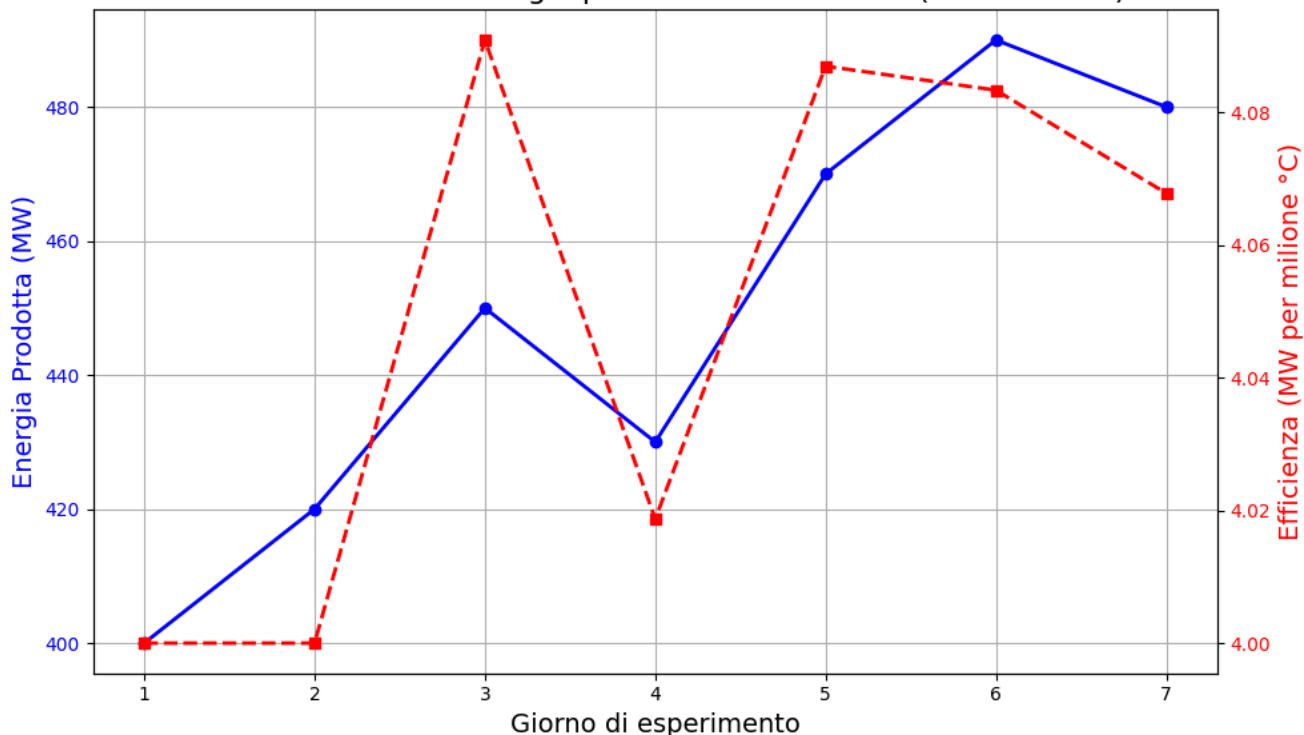
Tabella Dati ITER (7 dati):

	Giorno	Temperatura Plasma (milioni di °C)	Energia Prodotta (MW)	\
0	1	100	400	
1	2	105	420	
2	3	110	450	
3	4	107	430	
4	5	115	470	
5	6	120	490	
6	7	118	480	

Efficienza (MW per milione °C)

0	4.000000
1	4.000000
2	4.090909
3	4.018692
4	4.086957
5	4.083333
6	4.067797

Simulazione ITER: Energia prodotta ed Efficienza (introduzione)



Formula Efficienza Energetica (calcolo manuale):

Efficienza = Energia Prodotta (MW) / Temperatura Plasma (milioni di °C)

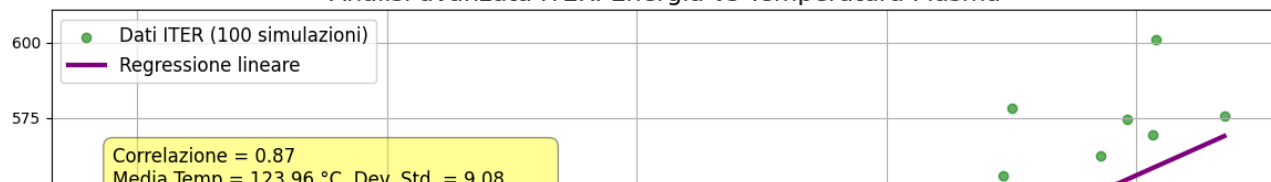
Tabella avanzata ITER (prime 10 righe su 100 dati):

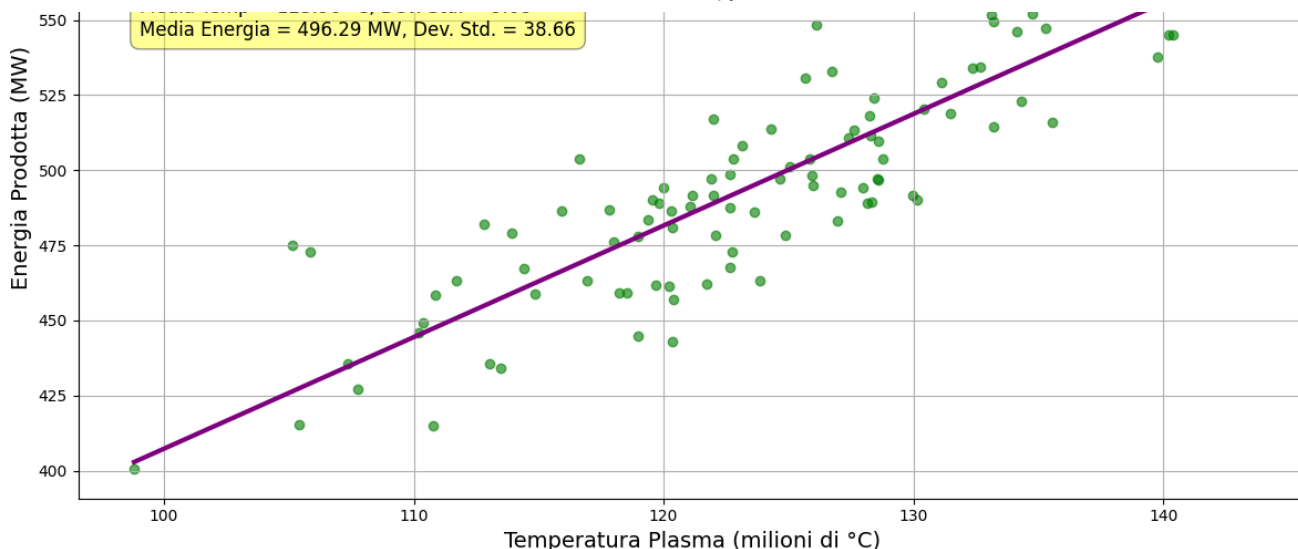
	Giorno	Temperatura Plasma (milioni di °C)	Energia Prodotta (MW)	\
0	1	129.967142	491.561151	
1	2	123.617357	486.056521	
2	3	131.476885	519.053251	
3	4	140.230299	544.875649	
4	5	122.658466	487.408151	
5	6	122.658630	498.715539	
6	7	140.792128	600.892231	
7	8	132.674347	534.188945	
8	9	120.305256	486.372032	
9	10	130.425600	520.213483	

Efficienza (MW per milione °C)

0	3.782196
1	3.931944
2	3.947867
3	3.885577
4	3.973702
5	4.065882
6	4.267939
7	4.026317
8	4.042816
9	3.988584

Analisi avanzata ITER: Energia vs Temperatura Plasma





## Formule Statistiche

Di seguito sono riportate tutte le formule utilizzate nelle analisi statistiche sul dataset ITER. Gli studenti possono usarle per eseguire calcoli manuali, consolidando così la comprensione dei concetti:

### 1. Media aritmetica ( $\mu$ ):

$$\mu = \frac{\sum_{i=1}^N x_i}{N}$$

Dove:

- $x_i$  sono i singoli valori osservati
- $N$  è il numero totale di osservazioni

### 2. Deviazione standard ( $\sigma$ ):

$$\sigma = \sqrt{\frac{\sum_{i=1}^N (x_i - \mu)^2}{N - 1}}$$

Dove:

- $\mu$  è la media
- $x_i$  i singoli valori osservati
- $N$  il numero di osservazioni totali

### 3. Covarianza (Cov):

La **covarianza** misura quanto due variabili cambiano insieme, indicando se un aumento (o diminuzione) in una delle due si accompagna a un aumento o diminuzione nell'altra. La formula è:

$$\text{Cov}(x, y) = \frac{\sum_{i=1}^N (x_i - \mu_x)(y_i - \mu_y)}{N - 1}$$

Dove:

- $x_i$  e  $y_i$  sono i singoli valori delle due variabili osservate (ad esempio, Temperatura e Energia)
- $\mu_x$  e  $\mu_y$  sono le rispettive medie delle due variabili
- $N$  è il numero totale delle coppie di osservazioni

#### Interpretazione:

- **Covarianza positiva:** le due variabili tendono a crescere o diminuire insieme.
- **Covarianza negativa:** una variabile cresce quando l'altra diminuisce, e viceversa.
- **Covarianza vicina a zero:** non c'è relazione evidente tra le due variabili.

#### 4. Correlazione (r):

La correlazione è la covarianza divisa per il prodotto delle deviazioni standard delle due variabili:

$$r = \frac{\text{Cov}(x, y)}{\sigma_x \cdot \sigma_y}$$

Dove:

- $\text{Cov}(x, y)$  è la covarianza appena vista
- $\sigma_x$  e  $\sigma_y$  sono le deviazioni standard delle due variabili

La correlazione è sempre compresa tra -1 e 1:

- **1** indica correlazione positiva perfetta
- **0** indica nessuna correlazione
- **-1** indica correlazione negativa perfetta

#### 5. Retta di regressione lineare:

L'equazione generale della retta è:

$$y = a + b \cdot x$$

Dove:

- $y$  è la variabile dipendente (Energia Prodotta)
- $x$  è la variabile indipendente (Temperatura Plasma)
- $a$  è l'intercetta (valore di  $y$  quando  $x = 0$ )
- $b$  è la pendenza (quanto cambia  $y$  al variare di  $x$ )

Nel nostro caso la retta di regressione calcolata è:

$$\text{Energia Prodotta (MW)} = 35.96 + 3.71 \times \text{Temperatura Plasma (milioni di } ^\circ\text{C)}$$

### ✓ Interpretazione della Regressione Lineare

La **regressione lineare** è uno degli strumenti fondamentali della Data Science. Serve a individuare e descrivere matematicamente la relazione che lega due variabili:

- **Variabile indipendente (x):**  
Temperatura del Plasma (milioni di  $^\circ\text{C}$ )  
(variabile che possiamo controllare o che osserviamo direttamente nel nostro esperimento)
- **Variabile dipendente (y):**  
Energia Prodotta (MW)  
(variabile che vogliamo prevedere o analizzare in relazione alla prima)

#### Cosa significa il valore della correlazione?

La **correlazione (r)** calcolata in questo esperimento simulato ITER è risultata pari a **0.87**.

Questo valore, vicino a **1**, indica che esiste una **forte relazione diretta e positiva** tra Temperatura del Plasma e Energia Prodotta:

- Quando la temperatura aumenta, l'energia prodotta tende ad aumentare di conseguenza.
- Una correlazione prossima a 0 significherebbe invece che non esiste alcuna relazione evidente tra le due variabili.
- Una correlazione negativa (vicina a -1) indicherebbe invece che al crescere di una variabile, l'altra diminuisce.

#### Interpretazione della Retta di Regressione Lineare

La retta di regressione calcolata ha questa equazione:

$$\text{Energia Prodotta (MW)} = 35.96 + 3.71 \times \text{Temperatura Plasma (milioni di } ^\circ\text{C)}$$

Dove:

- **35.96** (intercetta): rappresenta il valore teorico di energia prodotta quando la temperatura del plasma fosse teoricamente uguale a 0 (valore puramente matematico, poco realistico fisicamente, ma necessario per definire la retta).
- **3.71** (pendenza della retta): indica che per ogni incremento di 1 milione di  $^\circ\text{C}$  nella temperatura del plasma, l'energia prodotta aumenta mediamente di 3.71 MW.

#### Utilità Pratica della Regressione Lineare:

Grazie a questa formula possiamo prevedere facilmente quanta energia sarà prodotta da ITER conoscendo la temperatura del plasma. Ad esempio, se il plasma raggiungesse 130 milioni di  $^\circ\text{C}$ , potremmo calcolare così:

Per una temperatura di 130 milioni di °C, avremo:

$$\text{Energia prevista} = 35.96 + 3.71 \times 130 \approx 518 \text{ MW}$$

### Conclusione:

La regressione lineare ci permette di comprendere e visualizzare chiaramente le relazioni nascoste tra i dati, migliorare la nostra capacità predittiva e rafforzare decisioni scientifiche basate sui dati.

### Ora prova tu!

Scegli altri valori di temperatura e usa la formula per calcolare la relativa energia prodotta. Verifica così il potere predittivo del tuo modello statistico!

Inizia a programmare o [genera](#) codice con l'IA.

```
from google.colab import drive
drive.mount('/content/drive')
```

↗ Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force\_remount=True).

```
# Importo librerie
import pandas as pd
import matplotlib.pyplot as plt
```

```
# Link CSV diretto (cambia l'URL con quello tuo appena creato)
url_csv = "https://docs.google.com/spreadsheets/d/1hkgSkgN1Y209GJqDc0gWBuTSAiuKp2R16Ij199MSJhg/export?format=csv"
```

```
# Leggo il CSV direttamente online
df_sheets = pd.read_csv(url_csv, header=None)
```

```
# Controllo rapidamente struttura dati
print(df_sheets.head(30))
```

```
0 \
0 Data extracted on 01/07/2025 21:58:16 from [ES...
1 Dataset:
2 Last updated:
3 NaN
4 Time frequency
5 Type of plant
6 Operator/Trader
7 Energy balance
8 Standard international energy product classifi...
9 Unit of measure
10 NaN
11 TIME
12 GEO (Labels)
13 European Union - 27 countries (from 2020)
14 NaN
15 Special value
16 :
17 Observation flags:
18 p

1 \
0 NaN
1 Gross production of electricity and derived he...
2 24/06/2025 23:00
3 NaN
4 NaN
5 NaN
6 NaN
7 NaN
8 NaN
9 NaN
10 NaN
11 2019
12 NaN
13 80.567,258
14 NaN
15 NaN
16 not available
17 NaN
18 provisional
```

```
2 3 4 5 6 7 \
0 NaN NaN NaN NaN NaN
1 NaN NaN NaN NaN NaN
2 NaN NaN NaN NaN NaN
3 NaN NaN NaN NaN NaN
4 Annual NaN NaN NaN NaN
5 Total NaN NaN NaN NaN
6 Total NaN NaN NaN NaN
```

7	Gross electricity production	NaN	NaN	NaN	NaN	NaN
8	Solid biofuels	NaN	NaN	NaN	NaN	NaN
9	Gigawatt-hour	NaN	NaN	NaN	NaN	NaN
10		NaN	NaN	NaN	NaN	NaN
11		NaN	2020	NaN	2021	NaN
12		NaN	NaN	NaN	NaN	NaN
13		NaN	82.951,155	NaN	92.672,310	NaN
14		NaN	NaN	NaN	NaN	NaN

```
# Importo librerie necessarie
import pandas as pd
import matplotlib.pyplot as plt

# Link CSV diretto da Google Sheets
url_csv = "https://docs.google.com/spreadsheets/d/1hkgSkgN1Y209GJqDc0gWBuTSAiuKp2R16Ij199MSJhg/export?format=csv"

# Carico direttamente il CSV online senza intestazioni
df_sheets = pd.read_csv(url_csv, header=None)

# Indici precisi forniti da te
anni = df_sheets.iloc[11, [3, 5, 7, 9, 11]].astype(str).tolist()
produzione_biofuels = df_sheets.iloc[13, [3, 5, 7, 9, 11]].tolist()

# Conversione numeri europei in formato Python
produzione_biofuels = [
    float(str(val).replace(".", "").replace(",", ".")) if isinstance(val, str) else val
    for val in produzione_biofuels
]

# ITER simulato (attivazione dal 2023-2024)
produzione_iter_simulata = [0, 0, 0, 0, 5000]

# Grafico finale corretto chiarissimo
plt.figure(figsize=(10,6))

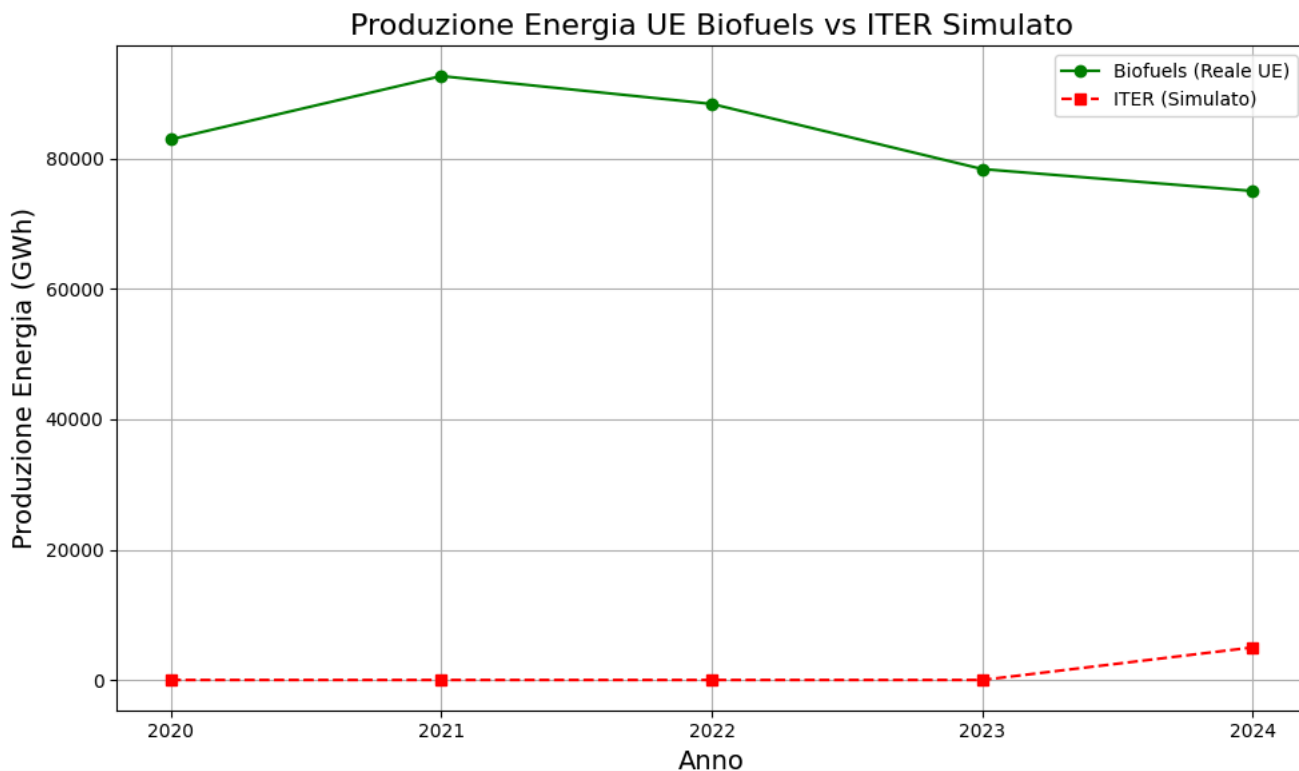
# Biofuels reali UE (verde)
plt.plot(anni, produzione_biofuels, marker='o', linestyle='--', color='green', label='Biofuels (Reale UE)')

# ITER simulato (rosso)
plt.plot(anni, produzione_iter_simulata, marker='s', linestyle='--', color='red', label='ITER (Simulato)')

# Impostazioni grafiche
plt.title("Produzione Energia UE Biofuels vs ITER Simulato", fontsize=16)
plt.xlabel("Anno", fontsize=14)
plt.ylabel("Produzione Energia (GWh)", fontsize=14)
plt.grid(True)
plt.legend()
plt.tight_layout()

# Mostro chiaramente il grafico finale completo
plt.show()
```





Inizia a programmare o [genera](#) codice con l'IA.

## Confronto tra produzione energetica reale (Biofuels UE) e simulata (ITER)

Questo grafico mostra chiaramente la produzione reale di elettricità nell'Unione Europea derivata dai **Biofuels solidi**, confrontata con la produzione energetica prevista (simulata didatticamente) dal reattore a fusione nucleare **ITER**.

### Considerazioni e riflessioni:

- La produzione da Biofuels mostra un andamento stabile con una leggera tendenza alla diminuzione negli ultimi anni.
- ITER, dal 2024, introduce la possibilità di produrre energia pulita attraverso la fusione nucleare, proponendosi come alternativa sostenibile alle fonti energetiche tradizionali.
- Quale impatto ambientale ed energetico potrà avere ITER nella transizione energetica europea?
- Come potrebbe evolvere il mix energetico europeo con l'integrazione della fusione nucleare?

Scrivi le tue riflessioni nel mini-report finale, valutando anche implicazioni etiche, economiche e sociali di queste nuove tecnologie energetiche.

## ✓ Procedura completa per scaricare dati reali da Eurostat, caricarli su Google Drive, convertirli in CSV e visualizzarli in Colab

Di seguito trovi una procedura semplice, chiara e guidata per studenti e formatori, utile per replicare la raccolta e analisi dei dati energetici reali europei e confrontarli con simulazioni didattiche (es. ITER).

### 1. Scaricare dati reali da Eurostat

1. Vai al sito ufficiale Eurostat: [Eurostat - Gross Electricity Production \(NRG\\_IND\\_PEHCF\)](#)
2. Scegli chiaramente questi filtri per ottenere dati coerenti e confrontabili:
  - **Geopolitical entity:** European Union - 27 countries (EU27)
  - **Energy balance:** Gross electricity production
  - **Unit of measure:** Gigawatt-hour (GWh)
  - **Energy product classification (SIEC):** ad esempio, Solid Biofuels
  - **Years:** ultimi 5 disponibili (es. 2019-2024)
3. Clicca su **"Download"** (in alto a destra) e scegli formato **"Excel (xlsx)"**.

### 2. Caricare il file Excel su Google Drive

1. Apri [drive.google.com](https://drive.google.com).
2. Premi "+ Nuovo → Caricamento file" e seleziona il file Excel appena scaricato.
3. Una volta caricato, clicca col tasto destro sul file e scegli "Apri con → Fogli Google".

Ora il file Excel è chiaramente disponibile come Foglio Google (Google Sheets).

### 3. Convertire Fogli Google in formato CSV leggibile da Python (Colab)

1. Dal foglio Google aperto scegli:
  - "File → Scarica → Valori separati da virgola (.csv)"
    - Verrà scaricato sul tuo PC un file .csv.
2. Se vuoi leggere il file direttamente online con Colab:
  - Premi "Condividi" in alto a destra.
  - Cambia le autorizzazioni del file in "Chiunque abbia il link può visualizzare".
  - Copia il link fornito.
  - Modificalo chiaramente così (sostituisci /edit?usp=sharing con /export?format=csv)

### 4. Visualizzare chiaramente i dati CSV in Colab (Python)

Usa il codice seguente nel notebook Colab per leggere e visualizzare chiaramente il file CSV direttamente online:

```
# Librerie necessarie
import pandas as pd
import matplotlib.pyplot as plt

# Inserisci chiaramente qui il tuo URL CSV diretto da Google Sheets
url_csv = "https://docs.google.com/spreadsheets/d/<ID>/export?format=csv"

# Leggi direttamente il CSV online
df = pd.read_csv(url_csv, header=None)

# Visualizza rapidamente le prime righe per capire la struttura
print(df.head(20))
```

Dopo aver eseguito, identifica chiaramente:

- La riga che contiene gli anni (es. 2019, 2020...).
- La riga con i dati numerici di produzione energetica.

Poi usa il codice finale adattato alle tue righe e colonne precise

### 5. Esercizio

Scarica altri dataset (es. Gas naturale, Carbone, Solare, Eolico) usando la stessa procedura.

- Confrontali chiaramente con la produzione simulata ITER.
- Scrivi una breve riflessione personale su vantaggi e sfide di ciascuna fonte energetica, considerando sostenibilità ambientale ed economica.

## ✓ Approfondimento dati reali (pwiDB - IAEA)

Per un approfondimento scientifico realistico sui processi di interazione plasma-materiale, puoi utilizzare dati sperimentali reali disponibili sul database ufficiale **pwiDB (Plasma-Wall Interaction Database)** gestito dall'IAEA (International Atomic Energy Agency).

Il database pwiDB raccoglie dati sperimentali sui principali processi fisici che avvengono nelle interazioni tra plasma e materiali, come:

- **Retention** (ritenzione)
- **Reflection** (riflessione)
- **Sputtering** (erosione da plasma)

### Come scaricare dati reali da pwiDB

1. Accedi al sito ufficiale pwiDB-IAEA:
  - [IAEA pwiDB \(Plasma-Wall Interaction Database\)](https://www.iaea.org/research-and-development/iaea/iaea-databases/pwiDB).
2. Seleziona chiaramente il processo desiderato, ad esempio:
  - **Process:** Retention
  - **Species:** H (idrogeno), Surface: seleziona chiaramente il materiale desiderato.