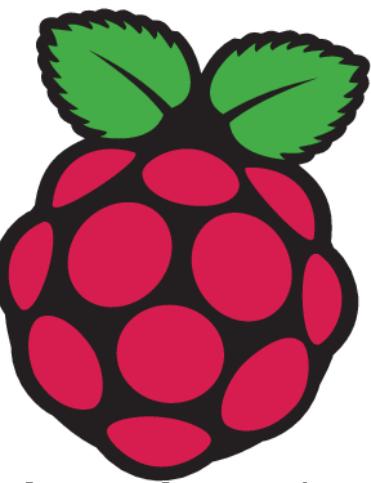




VISITA WWW.RASPBERRYITALY.COM

The MagPi



Numero 114 | Febbraio

2022 | [magpi.cc
raspberryitaly.com](http://magpi.cc/raspberryitaly.com)

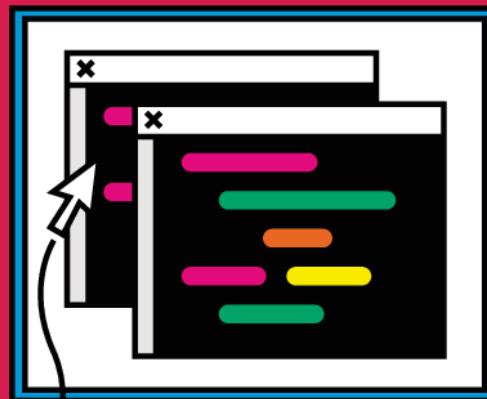
La rivista ufficiale Raspberry Pi
tradotta in italiano per RaspberryItaly

PROGRAMMAZIONE PRACTICA

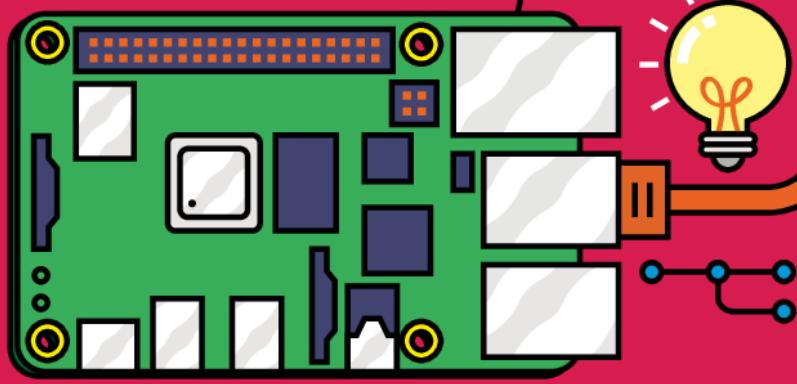
Crea le tue app, giochi e progetti!



Strumenti
Maker per
tutte le età



Monitorizza
le tue piante

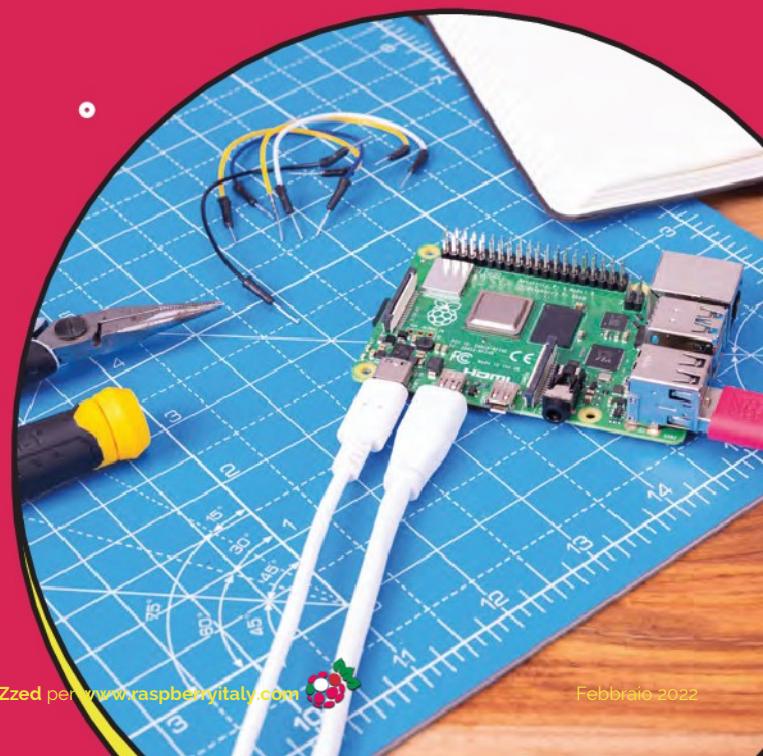


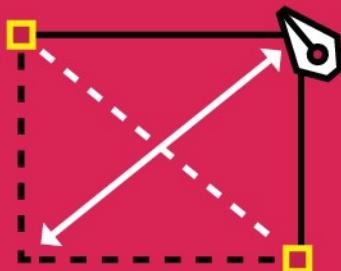
Estratto dal numero 114 di The MagPi. Traduzione di *marcolecce* e *Zzed*, revisione testi e impaginazione di Mauro "Zzed" Zolia (*zzed@raspberryitaly.com*), per la comunità italiana Raspberry Pi www.raspberryitaly.com. Distribuito con licenza CC BY-NC-SA 3.0. The MagPi magazine is published by Raspberry Pi (Trading) Ltd., Mount Pleasant House, Cambridge, CB3 0RN. ISSN: 2051-9982.



PROGRAMMAZIONE PRACTICA CON RASPBERRY PI

- Prendi confidenza con la programmazione e inizia a creare app, giochi e dispositivi. Di **Lucy Hattersley**



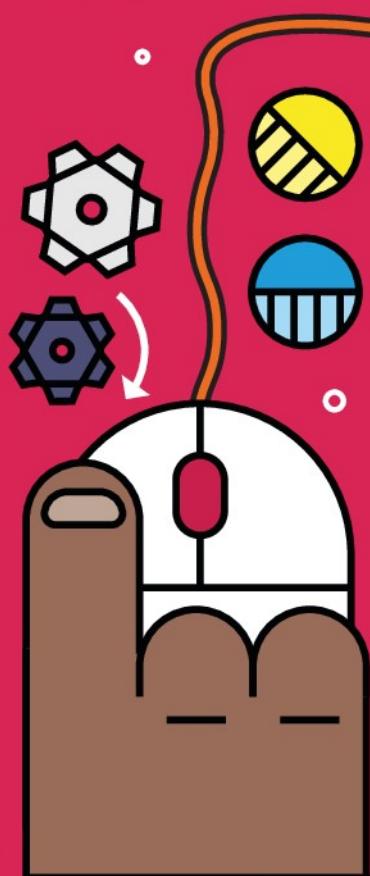
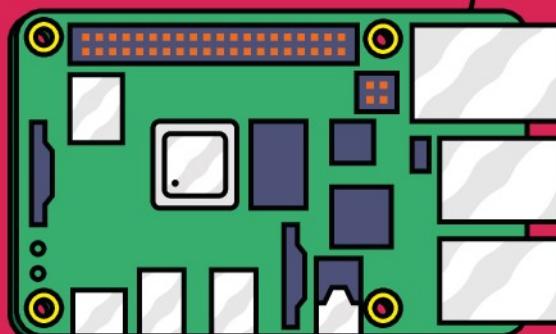


Ci siamo estremamente divertiti imparando a programmare con Raspberry Pi e nel 2022 la faremo diventare la nostra missione, per realizzare cose con il codice.

Se hai preso di recente un Raspberry Pi, siamo abbastanza certi che imparare a programmare sia in cima alla tua lista. O forse hai già una infarinatura di programmazione e sei alla ricerca di ispirazione. Ad ogni modo, questo speciale è qui per aiutarti.

In questo articolo, esamineremo tutte le risorse che abbiamo trovato utili durante l'apprendimento della programmazione nel corso degli anni e spiegheremo come è possibile programmare velocemente su Raspberry Pi. Tratteremo i concetti e gli strumenti chiave della programmazione e gli strumenti e le risorse che abbiamo trovato più utili. Infine, faremo un brainstorming con alcune grandi idee per app e giochi. E se vuoi fare le cose sul serio, ti mostriremo come fare a ottenere delle certificazioni e dimostrare di essere un programmatore.

Siamo convinti che il 2022 sarà un anno eccezionale! Quindi, prendiamo il nostro codice e facciamone buon uso. Realizziamo strumenti pratici con la nostra capacità di programmazione.



IMPOSTA IL TUO AMBIENTE

Tutto il necessario per iniziare un percorso di programmazione

Congratulazioni! Raspberry Pi OS è il migliore ambiente che puoi trovare per imparare a programmare. Quindi, sei partito bene. Raspberry Pi OS è una distro personalizzata basata su Debian Linux ed è ricca di ambienti di programmazione, strumenti utili, collegamenti a tutorial, progetti e aiuti online.

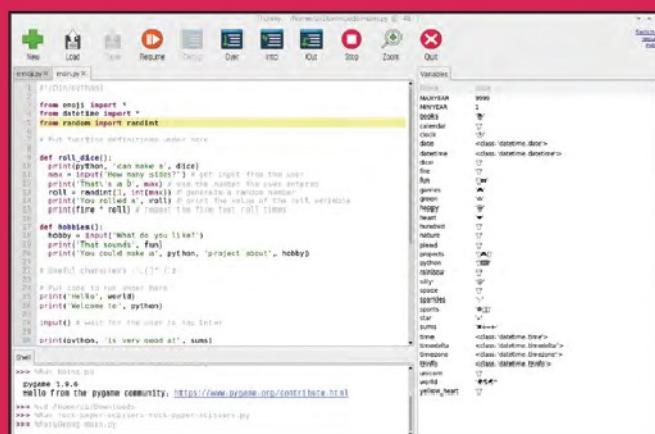
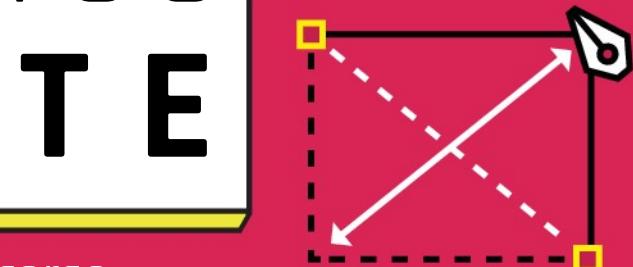
In questo articolo, utilizzeremo Raspberry Pi OS stock (installato tramite Imager, magpi.cc/imager). Se vuoi aiuto per configurare Raspberry Pi e Raspberry Pi OS, dai un'occhiata alla nostra Guida introduttiva alle Funzionalità Raspberry Pi su *The MagPi 113* (bit.ly/MagPi113It).

Fai clic sul menu delle applicazioni di Raspberry Pi e scegli Programmazione per trovare un sacco di software di programmazione preinstallato. La maggior parte delle persone programma utilizzando un IDE

- Thonny IDE evidenzia il codice in colori diversi, rendendo più facile la sua lettura

Caricare il codice in Thonny

Raspberry Pi OS viene fornito con una selezione di giochi creati per libro *Code the Classics* (magpi.cc/codetheclassics). Questi eleganti giochi sono tutti creati in Python e sono un buon esempio di ciò che puoi fare con la programmazione. Si trovano nel menu accessori sotto Giochi. Diamo un'occhiata al codice utilizzato per crearli. Apri Thony Python IDE e fai clic su Carica. Scegli Altre posizioni, fai clic su Computer e troverai i programmi sotto `usr/share/code-the-classics`. Carica `boing.py` e vedrai il codice per un gioco stile Pong, nella finestra principale. Fai clic su Esegui per lanciare il codice o su Debug se vuoi l'avanzamento a singoli step. Puoi ottenere una panoramica dettagliata di come funziona ogni programma, leggendo *Code The Classics*.



(ambiente di sviluppo integrato) e Thonny Python IDE è il tuo punto di partenza. In Thonny, programmerai in Python, che è il miglior linguaggio per i principianti; sono disponibili altri linguaggi di programmazione (ne parleremo tra un po').

Il vantaggio di utilizzare un IDE rispetto a un editor di testo risiede nelle sue funzionalità integrate che ti aiutano a digitare, eseguire il debug, ed eseguire il codice. La sintassi del testo è codificata tramite colori, semplificando la lettura e l'individuazione degli errori.

Se vuoi passare da Python ad altri linguaggi di programmazione, o sei pronto a altre dettagliate esperienze, allora l'IDE migliore è Visual Studio Code di Microsoft (o VS Code, magpi.cc/vscode). Il codice di Visual Studio è pieno di funzionalità avanzate, supporto a linguaggio esteso e le estensioni vengono utilizzate per fornire funzionalità aggiuntiva.

Da quando VS Code è arrivato su Raspberry Pi OS, abbiamo iniziato a usarlo come nostro ambiente di programmazione principale. Jim Bennett di Microsoft ha un blog su come ottenere Visual Studio Code attivo e funzionante su Raspberry Pi (magpi.cc/vscodeblog).



SCEGLI UN LINGUAGGIO

Parla il gergo imparando Il giusto linguaggio

Ci sono molti linguaggi di programmazione differenti tra cui scegliere e durante il tuo viaggio nella programmazione, in genere ne imparerai di diversi. La buona notizia è che più linguaggi di programmazione moderni condividono concetti simili: variabili, cicli if - while, definizione di funzioni, oggetti e così via. Quindi, una volta che hai imparato il primo linguaggio, il secondo è molto più semplice.

Oltre ai linguaggi di base, ci sono strumenti commerciali di cui avrai bisogno: database SQL, networking, implementazione della GUI e così via.

Quali linguaggi dovresti imparare è una conversazione in continua evoluzione, con nuovi linguaggi che arrivano e si evolvono ogni anno. Detto questo, ci sono alcune scelte che dovresti tenere d'occhio. Se sei un principiante assoluto, evita le chiacchiere e vai direttamente su Python.

Java

Molti programmi desktop sono costruiti in Java. Vale la pena saperlo, ma non iniziare con questo. Con tutta la sua utilità, Java è prolioso e complesso con molte complessità che fanno inciampare i nuovi arrivati. La buona notizia è che c'è una grande comunità; la cattiva notizia è che ne avrai bisogno.

Java era la lingua principale per lo sviluppo di app Android e ha un uso diffuso in tutto il settore industriale tecnologico. Quindi è sicuramente un buon strumento da avere nella tua cassetta degli attrezzi.

+ Utile, grande libreria di estensioni,
buono per lo sviluppo di app

- Crudo, contorto

@ magpi.cc/headfirstjava

Python

Python è il linguaggio di programmazione standard de facto per gli studenti. E per una buona ragione. La sua sintassi moderna rende il codice elegante e semplice da leggere e capire; Python è un linguaggio incredibilmente potente ed è usato da chi elabora dati e sviluppatori web, quindi non è solo un trampolino di lancio. È usato da Spotify, Facebook e YouTube. Puoi iniziare in piccolo: creare script e programmi da eseguire sul tuo computer. Python è molto popolare, quindi c'è un enorme supporto da siti Web, corsi e altri programmatore ed è stato adottato in tutto il settore.

+ Facile, potente, utile e ricco di tutorial

- Così bello che ti ci abitui

@ magpi.cc/pythonintro

C & C++

Il venerabile nonno dei linguaggi di programmazione. C è un linguaggio superbo e pensiamo che tutti dovrebbero imparare il C, ad un certo punto.

La maggior parte di Linux è costruito in C. Oltre al C esiste C++, un'estensione che racchiude in sé caratteristiche moderne come il supporto alla programmazione orientata agli oggetti. C (secondo noi) non è un ottimo modo per iniziare. Ma non è

+ Potente, ottimo per lo sviluppo desktop

- Complesso

@ magpi.cc/cgui

certo un brutto posto da visitare nel corso del tuo viaggio nella

programmazione e potresti decidere (come molti) che sia il posto dove vuoi sistemarti.

HTML & CSS

Vitale per creare siti web +
Non è un linguaggio completo -
magpi.cc/webdevbasics @

HTML (Hypertext Markup Language) e CSS (Cascading Style Sheets) sono i due linguaggi al cuore dei siti web. In genere non usi questi strumenti per creare dei programmi. Detto questo, entrambe sono abilità secondarie importanti che dovresti acquisire. Oltre a imparare a programmare, è una buona idea imparare come realizzare siti web. Puoi anche combinarlo con alcune abilità server, per ospitare il tuo sito web utilizzando software come Apache (magpi.cc/apache).

Potenzia i siti Web, abilità utile da avere +

Un po' limitato -

magpi.cc/introjavascript/magpi.cc/odinproject @

JavaScript

JavaScript è fondamentale per il web ed è (secondo StackOverflow, insights.stackoverflow.com) il linguaggio più popolare in uso oggi. Sebbene JavaScript sia utile per aggiungere potenza al tuo sito web, è più limitato di Python, Java o C. JavaScript è un'abilità utile da avere insieme a HTML e CSS, quindi dagli un'occhiata durante l'apprendimento delle abilità web.



CONCETTI BASE DI PROGRAMMAZIONE

Le cose di cui sono fatti i sogni



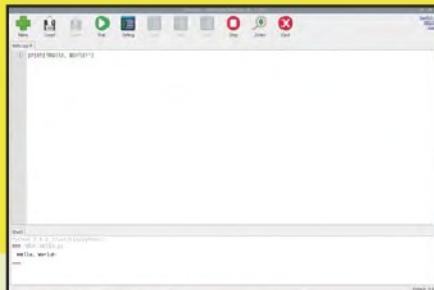
I tuo computer è configurato, il tuo IDE è aperto ed è tempo di imparare i concetti di base della programmazione. La buona notizia è che non devi imparare tanto per iniziare! È molto facile precipitare nella tana del coniglio della comprensione. Quindi fai attenzione a imparare di cosa hai bisogno e a tornare in carreggiata.

Ecco una breve panoramica dei concetti chiave che dovrà imparare, leggi il tutto per avere una panoramica, quindi inizia a cercare per saperne di più in ogni area. Ogni concetto ha una risorsa dedicata per saperne di più.

Sintassi

È tradizione iniziare qualsiasi linguaggio di programmazione ottenendo la scritta "Ciao Mondo!". Apri Thony Python IDE, fai clic su Nuovo e inserisci questa riga di testo nella finestra principale:

```
print("Ciao, Mondo!")
```



Fai clic su Esegui e Thony ti chiederà di salvare il file. Chiamalo "hello.py" e verrà eseguito. Vedrai: "Ciao, Mondo!" nella shell.

Sei partito bene. Ora devi conoscere la sintassi. Che cosa fa il comando **print?** E perché è **print** e non **Print**: perché è "Ciao, Mondo!" tra virgolette? E cosa sono quelle parentesi? Queste sono convenzioni che cambiano da linguaggio a linguaggio (anche se molte sono le stesse in qualsiasi linguaggio).

Vai al tutorial Python su W3Schools e completa la sezione sintassi (magpi.cc/pythonsyntax). Qualsiasi incontro con una nuova lingua dovrebbe iniziare con un corso sulla sua sintassi.



Variabili e dati

Una volta che ti sei esercitato a digitare e a eseguire programmi, la prima cosa che devi imparare sono le variabili. Queste sono contenitori per valori: numeri e stringhe di testo e così via.

```
foo = 32
bar = "Jill"
```

Quando scrivi questo, Python crea due variabili: **foo** contiene l'intero 32; **bar** contiene una stringa di caratteri che compongono il nome "Jill". Cosa sono **foo** e **bar**? Li troverai nei tutorial (o a volte vedrai **x** e **y**). Queste variabili possono essere qualsiasi cosa, si dice che **foo** e **bar** si riferiscono a un termine slang, ma vengono utilizzati solo come segnaposto. Le variabili possono essere qualsiasi parola che ti piace, e in genere il nome si riferisce al contenuto, come **first_name** o **age**. Ogni linguaggio ha le sue convenzioni, ma le variabili Python dovrebbero:

- Contengono solo caratteri minuscoli
 - Utilizzare i trattini bassi invece degli spazi
- Per una buona base sulle variabili, dai un'occhiata a Raspberry Pi Foundation's Introduction to Python (magpi.cc/pythonintro).



Struttura dei dati

Ora che stai creando variabili e sei alle prese con la sintassi di base, inizierai a pensare alle variabili come ai dati che compongono il tuo programma. Inizierai a raggruppare i dati in diversi gruppi. Questi elenchi di dati raccolgono un sacco di variabili in un unico spazio, che è esso stesso una variabile.

Di gran lunga il più comune è una semplice lista, che è un gruppo di elementi (tipicamente) simili. Quindi potresti avere tre stringhe chiamate Jack, John e Jill e una variabile elenco chiamata **persone**. Python ha sei diversi tipi di dati, ma hai bisogno solo di imparare liste, set e dizionari.

- LISTE.** Una sequenza di base di qualsiasi tipo di elementi (stringhe, numeri interi, ecc.) alla quale puoi aggiungere e rimuovere elementi in qualsiasi momento. Questo attributo è noto come "mutabile". Le liste sono comuni e sono facili da individuare in Python perché sono racchiuse tra parentesi quadre e le voci sono separate da virgole, ad es.:

```
nomi = ["Jack", "John", "Jill"]
```

- TUPLE.** Come le liste, ma una volta create non puoi aggiungere e rimuovere elementi. Le tuple sono scritte in Python con parentesi tonde:

```
nomi = ("Jack", "John", "Jill")
```

- SET.** Questi sono un po' più rari e difficili da comprendere. I set memorizzano

A Python list is a built-in data structure that holds a collection of items. To understand how lists work, let's go straight to the example. We want to create a list that includes names of the US presidents elected in the last six presidential elections. List elements should be enclosed in square brackets [] and separated by a comma. To create a list, we can simply write elements in square brackets:

```
us_presidents_list = ["George Washington", "John Adams", "Thomas Jefferson", "James Madison", "James Monroe", "John Quincy Adams", "Andrew Jackson", "Martin Van Buren", "William H. Harrison", "John Tyler", "Zachary Taylor", "Millard Fillmore", "Abraham Lincoln", "Ulysses S. Grant", "Rutherford B. Hayes", "James A. Garfield", "Chester A. Arthur", "Grover Cleveland", "Benjamin Harrison", "Grover Cleveland", "Theodore Roosevelt", "William Howard Taft", "Woodrow Wilson", "Warren G. Harding", "Calvin Coolidge", "Charles E. Hughes", "Franklin D. Roosevelt", "Harry S. Truman", "Dwight D. Eisenhower", "John F. Kennedy", "Lyndon B. Johnson", "Richard M. Nixon", "Gerald R. Ford", "Ronald W. Reagan", "George H. W. Bush", "Bill Clinton", "George W. Bush", "Barack Obama", "Barack Obama", "Donald Trump"]
```

The elements of this list appear in a specific order and start from the current president. If a president was elected twice (served for two terms), we include him twice. Here, we leverage two important characteristics of lists: lists are ordered and can contain duplicates.

When working with a list, you can iterate over it, access its elements, and also add, change, and remove list elements. These are the basic techniques that you can learn in our [PYTHON DATA STRUCTURES IN PRACTICE](#) course. Now, let's go through a couple of examples:

- Iterating over the list. By iterating over the list, we can perform a particular

diversi elementi in un'unica lista non ordinata. Ogni oggetto deve essere diverso, quindi non puoi per esempio, avere due Jill nella tua lista. Individuato dall'uso di parentesi graffe:

```
nomi = {"Jack", "John", "Jill"}
```

- DIZIONARI.** Più difficili da capire ma estremamente utili. I dizionari memorizzano i valori in coppie chiave:valore. Ogni oggetto ha una chiave e un valore. In genere li usi per ottenere il valore fornendo le chiavi. Individuato dall'uso di parentesi graffe e due punti di separazione tra le coppie chiave/valore. In questo esempio, i nomi sono la chiave e i valori sono l'età della persona":

```
nomi = {"Jack: 32", "Jill:29", "John:45"}
```

Tenere a mente le varie strutture dati può essere impegnativo. È meglio imparare liste, tuple e set e lasciare dizionari per dopo. Learn Python ha una buona pagina per aiutarti ad iniziare (magpi.cc/learnpythonlists).

SUGGERIMENTO! PEP 8

PEP 8 è la Guida di Stile per il coding in Python. Se ti stai chiedendo se una variabile dovrebbe essere in maiuscolo o quanti spazi utilizzare, consulta la guida di stile. Aggiungi questo ai preferiti!

> magpi.cc/pep8

Flusso

Dopo variabili e liste, è ora di cominciare a imparare il flusso del programma. Questa è la prima grande differenza tra un linguaggio come Python e un linguaggio di markup come HTML. Il flusso ti consente di prendere decisioni in un programma che utilizza istruzioni condizionali (in Python queste sono **if**, **elif** e **else**).

Puoi eseguire la stessa azione più volte, noto come "loop", usando le istruzioni **for** e **while**.

Per impararli, ti consigliamo di andare sull'introduzione a Python della Raspberry Pi Foundation (magpi.cc/pythonintro).

We use `when green flag clicked` at the time to make pen down. We could say if the pencil is blunt, then sharpen it. Similarly, `if` conditions let us write code that does something different depending on whether a condition is true or false.

`if (mouse is pressed?)` is needed for the target's code circle, and code to your mouse. An `if` condition to check if the `mouse is pressed?` is true or false.

An `if` statement for assignment - the `set [color v] of [pen v]` - set the value of a variable. An `if` statement for test equivalence - the `if [list v] is equal to [list v]` - if the things on either side have the same value, then the text is `true`, otherwise it is `false`.

Change the code in your `repeat []` to give a score:

```
when green flag clicked
repeat (5)
    if (mouse is pressed?)
        set [score v] to (score + 1)
    end
end
```

Text: Run your pencil. Try to stop that arrow on the blue outer circle to see your message. The colour of the pen at the centre of the arrow is the colour that is saved and checked.

The `if [true or false?]` controls how fast your game draws. If it's going too fast, wait for a few seconds.



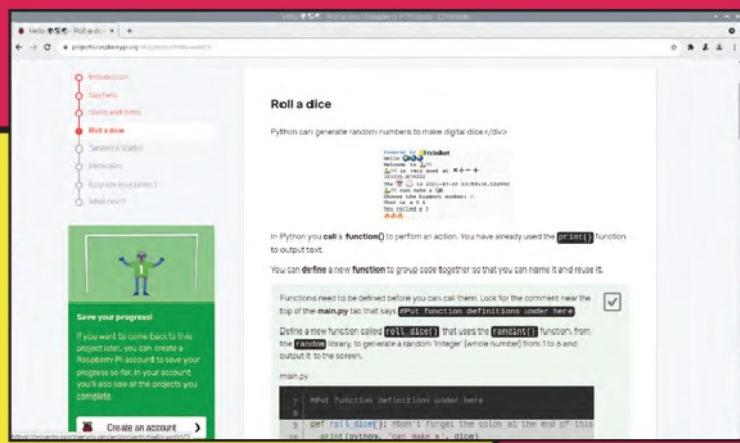
CONCETTI BASE DEL CODING



SUGGERIMENTO! TABULAZIONI O SPAZI

Un aspetto fondamentale del codice è l'indentazione. Sono spazi all'inizio di una riga. In Python, indenta con blocchi di quattro spazi e non usare mai il tasto tab.

> [magpi.cc/
pythonindentation](https://magpi.cc/pythonindentation)

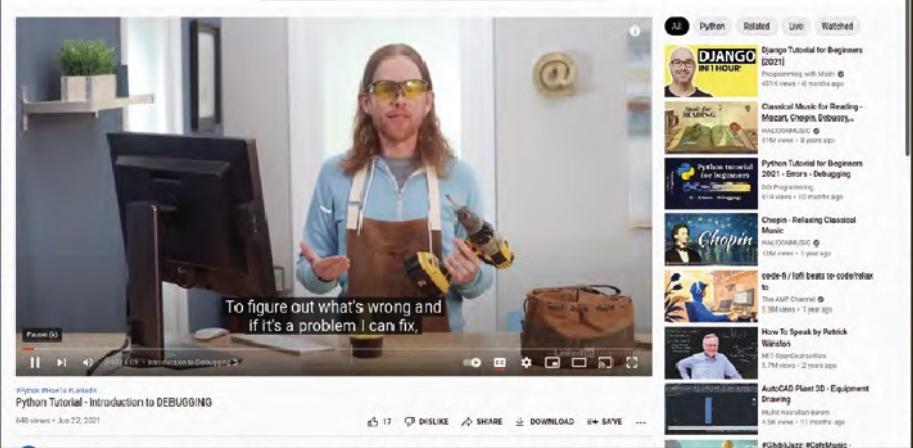


Funzioni

Ora che puoi creare variabili e controllare il flusso, è ora di conoscere le funzioni. Nei linguaggi di programmazione sono disponibili tutti i tipi di funzioni, come `print()` e `type()`. In Python, puoi individuarle come singole parole seguite da parentesi. Puoi trovare un elenco delle funzioni integrate in Python, qui: magpi.cc/pythonfunctions.

La vera potenza delle funzioni è creare un tuo pezzo di codice chiamato "definizione della funzione", che puoi individuare con `def` nel codice Python. Le definizioni delle funzioni sono come il copia e incolla per il codice. Invece di scrivere codice identico più volte, lo crei una volta, all'interno di una definizione di funzione e poi usi la funzione nel tuo programma.

Le funzioni sono una parte vitale della programmazione ed è fantastico creare le proprie. W3Schools ha un buon tutorial sulle funzioni in Python magpi.cc/w3pythonfunctions.

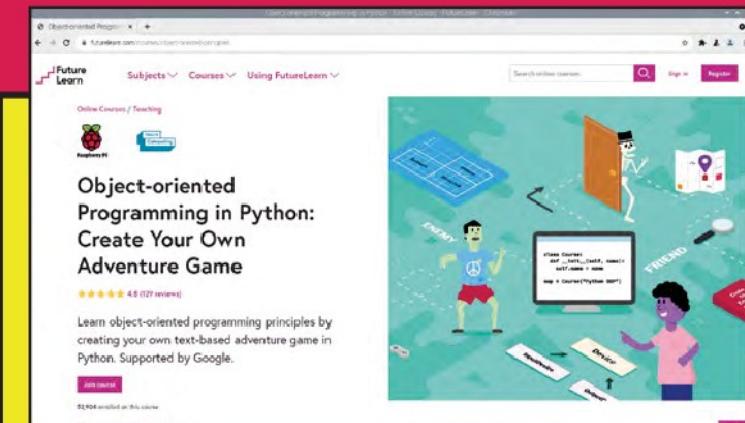


- “Abbiamo creato una guida alla programmazione orientata agli oggetti sul numero 54 di *The MagPi*”

Debugging

Questa è una cosa che non ti mancherà di praticare. Ogni programma che scrivi avrà bisogno del debug, a un certo punto. Poiché è qualcosa che ti impegnerà molto tempo, vale la pena imparare a farlo bene. Sasha Vodnik ha creato un grande video tutorial sul debug su LinkedIn Learning (magpi.cc/debugging) come parte di Programming Foundations: Beyond the Fundamentals. Dopodiché, freeCodeCamp ha una documentazione dettagliata su come migliorare le tue capacità di debugging (magpi.cc/debuggingskills).





Programmazione orientata agli oggetti

Una volta che hai preso confidenza con variabili e funzioni, puoi iniziare a riunire il tutto con la programmazione orientata agli oggetti (OOP). Questo è un concetto di programmazione complesso che raggruppa variabili e funzioni in "classi" e li usa per creare oggetti. Accederai poi alle funzioni, ora denominate "metodi" usando la notazione a punto. Confuso? Probabilmente lo sarai. Almeno all'inizio.

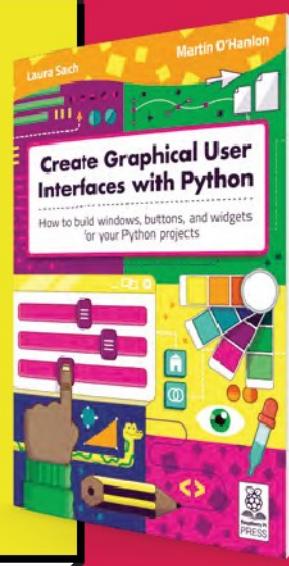
Imparare l'OOP è vitale, anche se è un argomento importante e fai attenzione a non farlo cadere nel dimenticatoio. Hai solo bisogno delle basi: come importare moduli e utilizzare la notazione a punto per accedere ai metodi. Questo ti consentirà di utilizzare il codice di altre persone nei tuoi programmi e accedere ad importanti API (interfacce di programmazione dell'applicazione).

Abbiamo creato una guida alla programmazione orientata agli oggetti nel numero 54 di **The MagPi** (bit.ly/MagPi54). Per un corso più pratico, dai un'occhiata al corso della Raspberry Pi Foundation e FutureLearn: 'Object-oriented Programming in Python: Create Your Own Adventure Game' (magpi.cc/futurelearoop).

GUI

Anche dopo aver appreso tutte queste abilità, lavorerai ancora principalmente su documenti di testo e riga di comando. Questo è un posto potente per un programmatore, ma la riga di comando non è dove trascorrerai la tua vita da non programmatore.

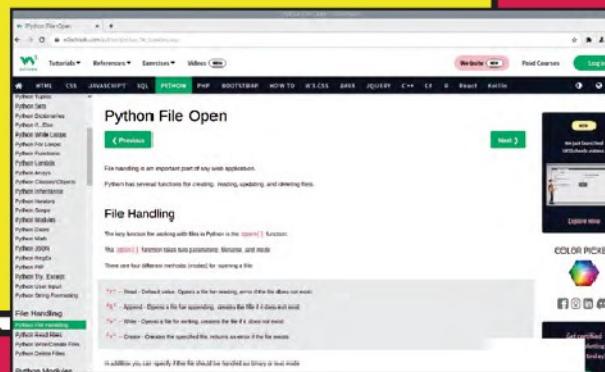
Per creare programmi funzionanti, avrai necessità di implementare una GUI (interfaccia grafica utente). Non preoccuparti, altre persone hanno fatto tutto il lavoro pesante e ci sono diverse API per diversi sistemi operativi. Ti consigliamo il nostro libro **Create Graphical User Interfaces with Python** (magpi.cc/gui).



Gestione dei file

Ad un certo punto, dovrà affrontare il salvataggio dei dati. Python gestisce tutti questi dati automaticamente, ma alla chiusura del programma, tutto viene ripulito e buttato.

Per salvare un programma, e usarlo più tardi, dovrà conoscere la gestione dei file. Avrai bisogno di salvare i dati nei file. Ancora, molti programmi si basano sulla gestione dei dati trovati in altri file (documenti di testo e immagini ed altro). W3Schools ha un buon corso sulla gestione dei file (magpi.cc/pythonfilehandling).



SITI DI FORMAZIONE

Ecco alcuni siti Web che offrono corsi di programmazione completi

FutureLearn

> magpi.cc/futurelearn

W3Schools

> magpi.cc/w3schools

Khan Academy

> magpi.cc/khan

freeCodeCamp

> magpi.cc/freecodecamp

MIT OpenSource

> magpi.cc/mitopen

Raspberry Pi Projects

> magpi.cc/projects

Coursera

> magpi.cc/coursera

Codecademy

> magpi.cc/codecademy

Udemy

> magpi.cc/udemy





SCEGLI UN PROGETTO

Inizia a programmare con uno scopo

I modo migliore per imparare e migliorare la tua abilità nella programmazione è quello di avviare un progetto. Puoi imparare molto dalla risoluzione dei problemi nei progetti e costruire qualcosa da zero. Poi, è semplicemente più divertente costruire qualcosa a cui sei personalmente interessato.

Quello che scegli dipende da te. Dipende dai tuoi interessi e ci sono molte idee in giro. Ti consigliamo di prendere una penna e iniziare a scrivere le idee, dopo aver superato le prime, le tue idee successive inizieranno a diventare interessanti.

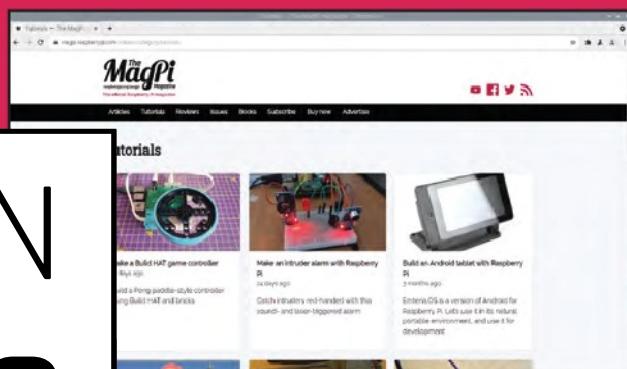
Ecco alcune aree per aiutarti a iniziare...

Costruire un sito web completo

Tutti usano il web e molti di noi hanno avuto esperienza di costruzione di siti web. Quindi, questo è un ottimo punto di partenza.

Naturalmente, poiché ora stai programmando e progettando per il web, puoi andare molto oltre e impostare un server e ospitarci il tuo sito web. Il corso freeCodeCamp Responsive Web Design è un buon posto per migliorare le tue abilità HTML e CSS (magpi.cc/responsiveweb). Web Dev Bootcamp di Udemy (magpi.cc/webdevbootcamp) fa un ulteriore passo avanti nelle applicazioni web e integrazione di database.

Costruire un server web o anche un browser web personale. Costruire un programma di web scraping è un divertente progetto di programmazione che ti insegna come eseguire la scansione dei siti Web. E puoi anche creare siti Web che integrano delle API, come un sito meteo che presenta il meteo in una posizione specifica. freeCodeCamp ha un video con sei di questi progetti web per aiutarti a iniziare (magpi.cc/6quickpython).

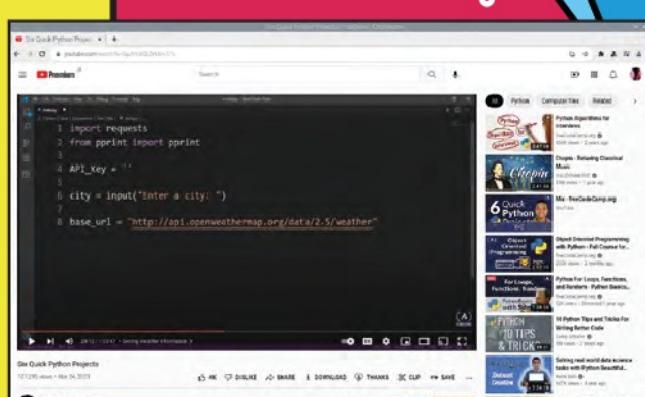


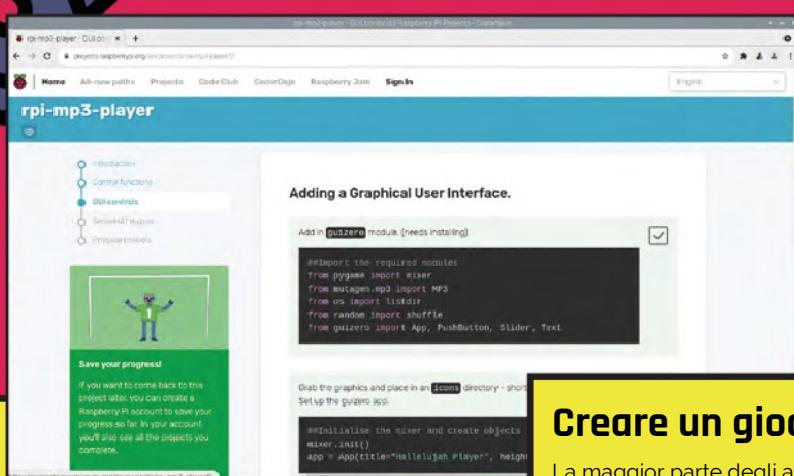
Creare qualcosa di fisico

Questa è un'area in cui abbiamo suggerimenti in abbondanza. Ogni mese all'interno di questa rivista troverai progetti e tutorial che combinano l'hardware elettronico di Raspberry Pi con il mondo reale. Raspberry Pi è l'ideale per la creazione di oggetti del mondo reale.

Vai al nostro sito Web e fai clic su Tutorial (magpi.cc/tutorials). In questo momento, ci sono guide per la realizzazione di un tablet Android, un allarme anti-intrusione, un segnale digitale non disturbare, e un timer Pomodoro. Inoltre, altre centinaia di idee. Che ne dici di prendere l'ultima copia di **Raspberry Pi Handbook 2022** (magpi.cc/handbook2022)? Ha oltre 200 pagine piene di progetti da provare a casa tua.

Se vuoi aiuto per iniziare con l'elettronica, dai un'occhiata alla guida Get Started with Electronics di Mark Vanstone (magpi.cc/electronics), ricco di informazioni sui circuiti di prototipazione con breadboard e aggiunta di batterie, pulsanti, interruttori, altoparlanti, e tutti i tipi di widget per il tuo progetto.





Costruire un'applicazione

Creare siti Web è divertente, ma molte persone imparano a programmare per creare applicazioni autonome, sia per il desktop che come app per dispositivi mobili.

Entrambe sono assolutamente possibili con Raspberry Pi. Tanto che la vera sfida è scegliere un progetto di programma da sviluppare. Nel processo di apprendimento del codice, costruirai molte applicazioni a riga di comando, ma integrarle in una GUI ti aiuterà a passare alla creazione di un programma completo (magpi.cc/gui). **Create Graphical User**

Interfaces with Python ti insegnerebbe come creare una semplice applicazione di disegno e un creatore di GIF animate. Al di là della libreria guizero, troverai molte utili librerie Python. Guizero è realizzata a partire da Tkinter (magpi.cc/tkinter), e per il 20% che non puoi ottenere con guizero, usa Tkinter. Oltre a queste, c'è PyQt5 (magpi.cc/pyqt5) che fornisce ulteriori collegamenti per funzioni come posizione, NFC, e Bluetooth ed estende lo sviluppo ad altre piattaforme comprese le app mobili.

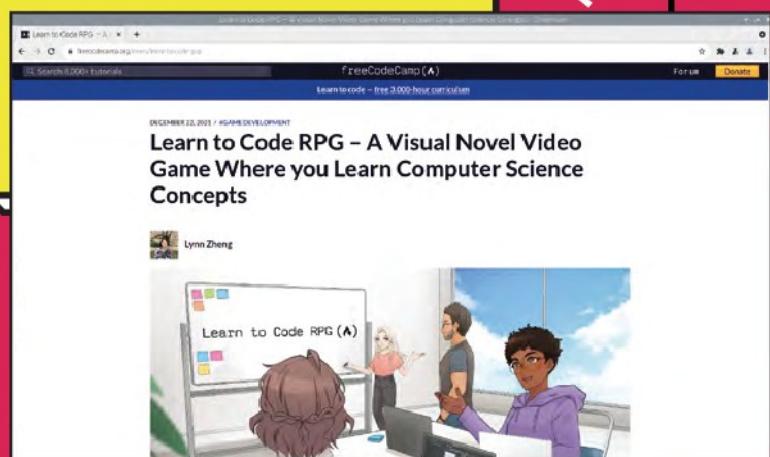
La cosa più importante è scegliere un'idea. Prendi carta e penna e continua a fare brainstorming fino a trovare qualcosa che ti attiri. Potresti creare un lettore musicale, un aggregatore di notizie, monitoraggio delle spese, app per l'elenco delle cose da fare, un editor di immagini, o qualsiasi cosa tu voglia.

“ Creare un gioco è un buon modo per allenare le abilità di programmazione con maggiore libertà di espressione ”

Creare un gioco

La maggior parte degli appassionati di computer è cresciuta con i videogiochi, in una forma o nell'altra, e creare un gioco è un buon modo per allenare le abilità di programmazione con la maggiore libertà di espressione. Il mercato dei videogiochi indie non è mai stato così vivace, quindi una buona storia potrebbe attrarre l'attenzione del pubblico.

Se hai un ricordo dei videogiochi degli anni '80, **Code the Classics** è il punto di partenza (magpi.cc/codetheclassics). È dotato di riedizioni di videogiochi di stile classico, come Pong e Pac-Man, ma codificati con Pygame Zero (magpi.cc/pygamezero). La nostra rivista sorella dal titolo Wireframe (wfmag.cc) ha una sezione mensile di codice sorgente che contiene lo sviluppo di videogiochi. Learn to Code RPG è un nuovo gioco open-source con oltre 600 quiz di informatica. Se vuoi creare un gioco di ruolo, dai un'occhiata a **Make an Adventure Game with Ren'Py** di Mark Vanstone (magpi.cc/makeadventure) o **Make Your Own Video Games feature** di KG Orphanides su **MagPi** 73 (bit.ly/MagPi73It), che ti guida attraverso il processo di creazione e distribuzione dei giochi.



TESTA LE TUE ABILITÀ

Vuoi mostrare cosa Hai ottenuto? Dai un'occhiata a questi siti relativi al codice.

HackerRank

> magpi.cc/hackerrank

Codewars

> magpi.cc/codewars

LinkedIn Learning

> magpi.cc/linkedinlearning

edX

> magpi.cc/edx

FutureLearn

> magpi.cc/futurelearn



Phil King

MAKER

Collaboratore di *The MagPi* di lungo corso. Phil è uno scrittore freelance e un editore con un focus sulla tecnologia @philkingeditor

Cosa Serve

- Sensore di umidità magpi.cc/moisturesensor
- Sensore livello liquido magpi.cc/liquidlevel
- MCP3008 ADC magpi.cc/mcp3008
- Ponticelli

- Le notifiche Pushbullet possono essere inviate a smartphone o al browser Chrome attivo su un altro computer

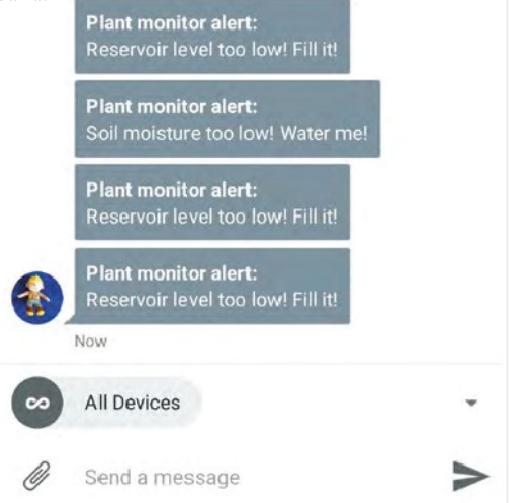
Mondo Sensori: Monitorizza le tue piante

Controlla l'umidità del terreno della tua pianta e invia un avviso se è troppo secca

In questa serie, stiamo esplorando alcuni dei sensori più comuni e i loro utilizzi. In precedenza, abbiamo costruito un paio di allarmi – uno per la sicurezza antincendio e del gas, l'altro per rilevamento di intrusi – e una stazione meteorologica di base.

Nell'ultimo articolo, abbiamo utilizzato un ADC per leggere l'uscita analogica di un sensore UV. Questa volta utilizzeremo un ADC per controllare il valore di umidità di un sensore posto in un vaso per piante. Aggiungeremo anche un sensore di livello per controllare il livello dell'acqua in un serbatoio (sottovaso) dove è posto il vaso della pianta.

Se il terreno o il serbatoio dell'acqua sono asciutti, il monitor ci invierà un avviso Pushbullet dicendoci di annaffiare la pianta per mantenerla sana.



01 Collegare l'ADC

Poiché i due sensori che useremo per il monitor dell'impianto forniscono entrambi le loro letture come segnale analogico, dovremo convertirlo in un valore digitale utilizzando un chip ADC (analogue-to-digital converter). Come in passato, stiamo usando l'ADC MCP3008, che ha otto canali di ingresso.

Poiché il chip ADC utilizza l'interfaccia SPI, dovremo abilitarla nel tool Configurazione Raspberry Pi. È anche meglio abilitare il supporto completo di SPI in Python 3. Per farlo, apri una finestra di Terminale e inserisci:

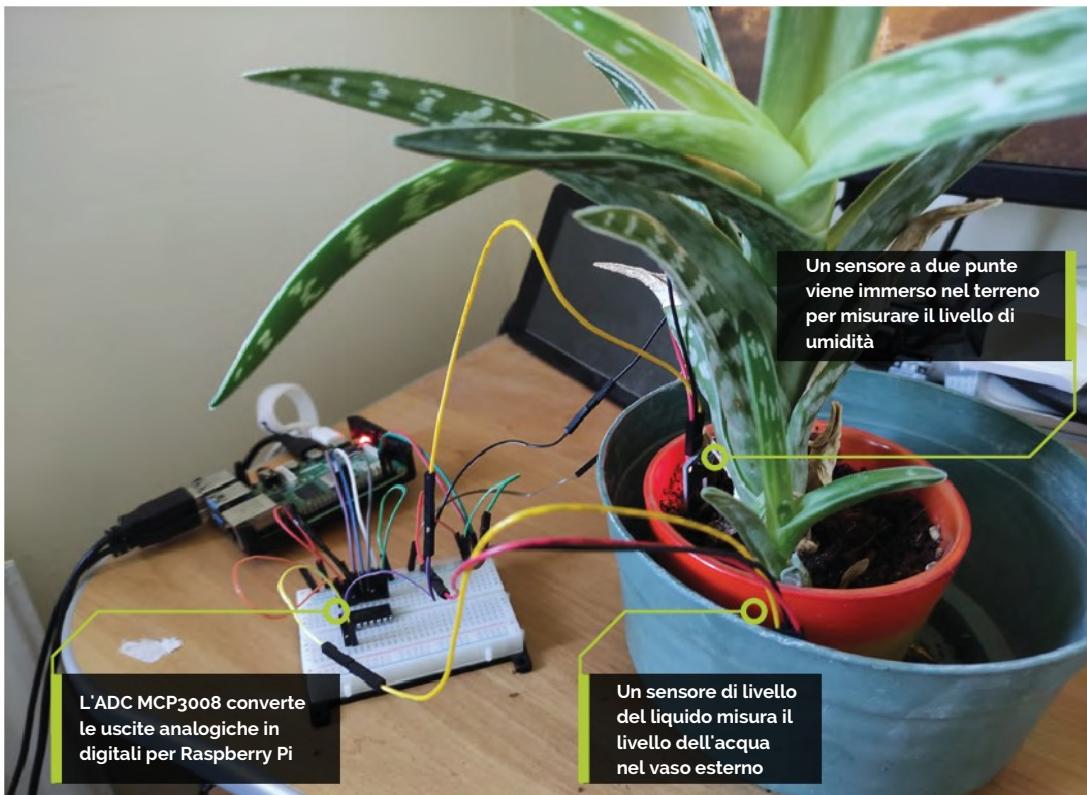
```
sudo apt-get install python3-spidev
```

Ora, con l'alimentazione di Raspberry Pi spenta, è ora di collegare il nostro ADC: potresti averlo già fatto dall'ultima puntata di questa serie. Inserisci l'MCP3008 al centro della breadboard, a cavallo del suo solco centrale. Assicurati che sia inserito nel modo corretto, come mostrato nello schema di cablaggio della **Figura 1**, con la parte superiore della scritta sopra l'ADC più vicino al Raspberry Pi.

Ora collega i ponticelli come in **Figura 1**. Due vanno al binario di alimentazione della breadboard "+", collegato a un pin 3V3; altri due sono collegati a un pin GND tramite il binario "-". Le quattro gambette centrali dell'ADC sono collegate alla interfaccia SPI sul Raspberry Pi: pin GPIO 8 (CEO), 10 (MOSI), 9 (MISO) e 11 (SCLK).

Ricontrolla che siano tutti collegati ai pin giusti altrimenti non funzionerà correttamente e potresti persino danneggiare il chip ADC o il tuo Raspberry Pi.





01 Ricontrolla che siano tutti collegati ai pin giusti altrimenti non funzionerà correttamente **01**

02 Collegare il sensore di umidità

Con l'ADC collegato correttamente al Raspberry Pi, ora possiamo aggiungere il nostro sensore di umidità alla configurazione. Ne stiamo usando uno dal Waveshare Sensors Pack disponibile nel Regno Unito da The Pi Hut (magpi.cc/wavesensors). Il sensore è disponibile anche separatamente, e potresti usare anche sensori di umidità del terreno alternativi.

Come in **Figura 1**, collegiamo il pin VCC del sensore a 3,3 V tramite il binario di alimentazione della breadboard e il suo pin GND al pin GND del Raspberry Pi tramite i contatti della breadboard.

Dal momento che vogliamo posizionare il sensore in un vaso, stiamo usando un ponticello femmina/femmina lungo (come quelli forniti nel pacchetto sensori Waveshare) per coprire la distanza tra il nostro Raspberry Pi e la breadboard.

Infine collegiamo l'AOUT del sensore (uscita analogica) sul pin canale 0 dell'MCP3008, come mostrato nella **Figura 1**. È possibile collegarlo a uno qualsiasi degli otto canali su quel lato, ma stiamo usando questo, nel nostro codice.



03 Testare il livello di umidità

Creiamo un semplice programma per testare il sensore. Nel listato **moisture_test.py**, stiamo utilizzando la libreria GPIO Zero in quanto ha una pratica classe **MCP3008**, che importiamo all'inizio. Assegniamo la variabile `moisture` al canale 0 dell'MCP3008 per leggere l'uscita analogica del sensore collegato.

In un loop `while True:`, moltiplichiamo digitalmente l'uscita convertita del sensore (che varia da 0 a 1) per la tensione massima di 3300 (in microvolt) per ottenere una lettura accurata. Nella nostra istruzione `print`, il parametro `%-3.2f` imposta ogni output con un minimo di tre cifre di cui due decimali.

Top Tip

Sensore digitale

Abbiamo usato un sensore di umidità con uscita analogica, ma alcuni offrono un'uscita digitale, come ad esempio questo: magpi.cc/soilsensor.

Figura 1 Lo schema di cablaggio per l'impianto di monitoraggio, compresi il sensore di livello del liquido, sensore umidità del suolo e ADC

Top Tip

Aggiungere un display

Per vedere a colpo d'occhio come sta la tua pianta, potresti aggiungere un mini display LCD al tuo Raspberry Pi per mostrare le letture dei sensori.

Puoi modificarlo in base alle tue preferenze. Noi aggiungiamo il parametro `end = "\r"` in modo che il messaggio venga sempre stampato sulla stessa riga.

Se il sensore non è posizionato, la lettura dovrebbe essere molto bassa (la nostra era di 1,61 mV). Per testarlo, prova a tenerlo in mano e la lettura dovrebbe salire a circa 200 mV o più, a seconda su quanto sono umide le tue dita - agiscono come un conduttore tra i due poli del sensore in modo che la corrente passi dall'uno all'altro. Più il materiale è umido e più è conduttivo. Questo vale anche per il terreno in un vaso per le piante.

Se inumidisce le dita e provi il test di nuovo, dovrà vedere la lettura aumentare fino a 800 mV. Fai molta attenzione a non bagnare l'elettronica, però.

04 Umidità del vaso

Ora sappiamo che il sensore di umidità sta funzionando correttamente, proviamo a metterlo in un vaso per le piante. Mettiamo il nostro in un terreno abbastanza asciutto per prendere un

moisture_test.py

Linguaggio: Python 3

```
001. from gpiozero import MCP3008
002.
003. moisture = MCP3008(0)
004.
005. while True:
006.     print("Umidità: %-3.2f mV    % ("
            3300 * moisture.value), end = "\r")
```

liquid_test.py

Linguaggio: Python 3

```
001. from gpiozero import MCP3008
002.
003. liquid = MCP3008(1)
004.
005. while True:
006.     print("Livello del liquido: %-3.2f mV    % ("
            3300 * liquid.value), end = "\r")
```

riferimento di lettura. Saremo quindi in grado di usarlo per attivare un avviso quando il terreno diventa troppo secco e la pianta ha bisogno di essere annaffiata.

Eseguendo nuovamente il programma `moisture_test.py`, abbiamo scoperto che il terreno asciutto dava una lettura ben al di sotto di 100 mV. Quando abbiamo annaffiato la pianta, la lettura è cresciuta oltre i 1400 mV.

05 Sensore livello liquido

Se hai messo il tuo vaso in un sottovaso o portavaso contenente un serbatoio d'acqua, puoi anche aggiungere un sensore di livello del liquido per controllare il livello dell'acqua. Questo è un passaggio facoltativo, e probabilmente un po' eccessivo, il sensore di umidità dovrebbe essere sufficiente.

Il nostro sensore di livello del liquido proviene dal Waveshare Sensors Pack ma è venduto anche separatamente. Funziona più o meno allo stesso modo del sensore di umidità, con l'acqua che conduce una corrente tra le strisce di metallo. Più alto è il livello dell'acqua, più alta è l'uscita in tensione: da circa 0 V a 0 cm a 1,88 V a 4,8 cm. Quindi possiamo usarlo per attivare un avviso quando il livello dell'acqua scende quasi a zero.

Collegalo come in Figura 1, con il pin AOUT collegato al pin del canale 1 dell'ADC. Provalo con il listato `liquid_test.py` e immergilo a diversi livelli nell'acqua per vedere la tensione modificarsi. Ancora una volta, fai attenzione a non bagnare le connessioni elettroniche o il Raspberry Pi.

 **Mettiamo il nostro in un terreno abbastanza asciutto per un riferimento di misurazione** 

06 Notifiche push

Scriveremo un programma per leggere i due sensori e attivare un avviso quando uno dei due, l'umidità o il livello del liquido, è troppo basso.

Sembra che potremmo far suonare un cicalino, come per gli allarmi delle parti 1 e 2 di questa serie, abbiamo pensato che sarebbe stato più utile inviare una notifica push a un telefono o computer. Per questo, stiamo utilizzando Pushbullet, che offre un livello di servizio gratuito.

Vai su pushbullet.com e registrati un account. Quindi vai su Settings > Account sul sito Web e fai clic su Crea token di accesso. Copialo, poiché dovrai aggiungerlo al codice.

Dovremo anche installare la libreria Python 3 per Pushbullet. Dopo esserti assicurato che il Raspberry Pi sia aggiornato, apri un Terminale e inserisci: `sudo pip3 install pushbullet.py`.





▲ I due sensori che stiamo usando: umidità (in alto) e livello del liquido (in basso)

07 Codice monitoraggio piante

Nel nostro codice finale, `plant_monitor.py`, importiamo la libreria `pushbullet` e assegniamo la variabile `pb` al nostro account Pushbullet – dovrà sostituire il `Your Access Token` con il token di accesso pushbullet ottenuto nel passaggio precedente. Avrai anche bisogno del nome del tuo Dispositivo-Pushbullet connesso a cui vuoi inviare la notifica push. È possibile trovare i nomi dei dispositivi sia sul sito web Pushbullet, in Settings >Devices, o utilizzare la linea `print(pb.devices)`, come mostrato nella parte superiore del codice.

Nella funzione `alert`, impostiamo la variabile `device` con il nome del dispositivo a cui inviare la notifica. Quindi inviamo una notifica push con il testo "Monitoraggio Piante:" seguito dalla stringa di `message` determinata dall'istruzione condizionale nel loop `while True`. Aggiungiamo uno `sleep` di 60 secondi (o più se vuoi) in modo che più notifiche non vengono inviate una subito dopo l'altra.

Nel ciclo `while True`: controlliamo se le letture per l'umidità e il livello del liquido (in mV), le cui soglie fissate rispettivamente a 500 e 1000 (modifica in base alle tue preferenze), siano o meno sotto i limiti ed eventualmente viene attivato un avviso con il relativo messaggio.

Andare oltre

Ora abbiamo un monitoraggio delle piante funzionante che ci avvisa quando c'è bisogno di annaffiare. Fai un passo ulteriore creando un sistema per irrigare la pianta automaticamente quando il terreno diventa troppo asciutto. Per questo, potresti usare un'elettronovalvola per aprire e chiudere l'approvvigionamento idrico o una pompa dell'acqua. Avrai bisogno anche di una scheda relè per controllare il solenoide o la pompa, poiché probabilmente avrà una tensione e una corrente più alta di quella che Raspberry Pi può fornire.

plant_monitor.py

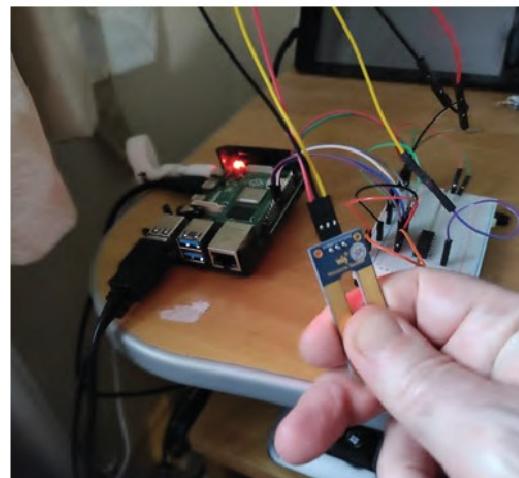
SCARICA IL CODICE COMPLETO:

magpi.cc/github

```

001. from gpiozero import MCP3008
002. from time import sleep
003. from pushbullet import Pushbullet
004.
005. pb = Pushbullet("Your Access Token")
006. print(pb.devices)
007.
008. message = ""
009. moisture = MCP3008(0)
010. liquid = MCP3008(1)
011.
012. def alert():
013.     device = pb.get_device('Il tuo Dispositivo')
014.     push = device.push_note("Allerta Monitoraggio Piante: ", message)
015.     sleep(60)
016.
017. while True:
018.     moisture_mv = 3300 * moisture.value
019.     liquid_mv = 3300 * liquid.value
020.     print("Umidità %-3.2f mV    " % moisture_mv,
021.           "Livello del liquido: %-3.2f mV    " % liquid_mv, end = "\r")
022.     if moisture_mv < 500:
023.         message = "Umidità del terreno troppo bassa! Innaffiami!"
024.         alert()
025.     elif liquid_mv < 1000:
026.         message = "Livello del serbatoio troppo basso! Riempilo!"
027.         alert()

```



▲ Testare il sensore di umidità tenendolo in mano; la corrente passa da un polo all'altro

Top Tip

Guardala crescere

Collega un Camera Module al tuo Raspberry Pi e fai una foto alla pianta ogni dieci minuti, quindi combinale in un video time-lapse per vedere come cresce.



STRUMENTI DA MAKER

PER TUTTE LE ETA'

Kit indispensabile per costruire in casa

Se hai un Raspberry Pi, hai già fatto un passo nel making. La programmazione è un ottimo modo per divertirsi e imparare Raspberry Pi, puoi anche usarla per controllare cose nella vita reale.

Avrai però bisogno di alcuni strumenti e materiali per il lavoro. Non tutti sono adatti a tutti i gruppi di età, il che può causare un po' di mal di testa quando stai cercando di inventare delle cose da realizzare.

Non temere, abbiamo la soluzione per te. Nelle prossime pagine, illustreremo le tipologie di strumenti sicuri sia per i maker più giovani che per quelli più anziani. Preparati a costruire una tempesta.





IL TUO SPAZIO MAKER

Dove dovrà costruire?

01 Trova il tuo spazio

Progetti diversi richiedono tipi diversi di spazio. Non tutti hanno accesso a un garage o capannone, ma potresti avere una scrivania o un tavolo. Procura tappetini da taglio e assi di legno su cui lavorare se non vuoi danneggiare la superficie che stai utilizzando e assicurati che qualunque altra cosa che non vuoi danneggiare, sia stata allontanata.

02 Attenzione alla sicurezza

Devi assicurarti che qualsiasi spazio tu scelga sia sicuro da utilizzare, sia per te che per chiunque altro. Probabilmente lavorerai con strumenti che possono surriscaldarsi o che sono taglienti, quindi tieni gli oggetti infiammabili lontano, indossa occhiali dove necessario e indossa guanti spessi se ne hai bisogno. Taglia lontano dal corpo e assicurati che tutti gli strumenti siano fuori dalla portata di bambini e amici pelosi.



03 Immagazzinare le idee

Questo può far inciampare alcuni nuovi maker – dove terrai i tuoi strumenti o progetti in corso? I cassetti in plastica sono economici e robusti per strumenti e simili, oppure potresti inserirli in una cassetta degli attrezzi. Per progetti più piccoli, puoi usare facilmente un cassetto, ma con quelli più grandi devi considerare come influenzereanno chiunque altro vicino al tuo spazio maker.



Attenzione! Saldatura

I saldatore diventano molto caldi e rimangono tali per molto tempo dopo essere spenti. Documenta sulla saldatura prima di collegare i fili
magpi.cc/soldering

Attenzione! Utensili elettrici

Gli utensili elettrici richiedono qualche istruzione e di allenarsi in modo corretto per l'uso, così come attrezzatura di sicurezza
magpi.cc/drillsafety

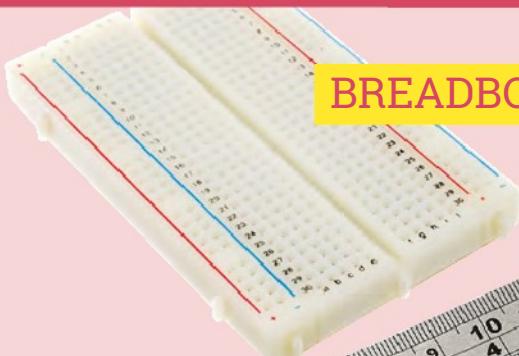
Attenzione! Lame

Puoi tagliarti facilmente con le lame affilate. Taglia lontano dal corpo e resta al sicuro

STRUMENTI PER GIOVANI MAKER

Tutti devono iniziare da qualche parte. Anche se alcuni di noi potrebbero essere stati in grado di usare trapani a percussione sin dalla giovane età, non ne abbiamo sempre posseduto uno. Ecco il tipo di cose che un giovane maker dovrebbe essere in grado di usare in sicurezza

STRUMENTI



BREADBOARD

METRO IN METALLO

Spelare fili con le forbici

Le forbici possono essere usate per tagliare il filo, ma lo sapevi che puoi usarle anche per togliere parte dell'isolamento plastico dal filo? Lo spezzone di filo deve essere spelato per esporre il filamento di metallo interno, in modo da poterlo inserire in una breadboard – basta posizionare le forbici sul filo a circa un centimetro o due dal taglio, applicare una leggera pressione e torcere su se stesso il filo. Tira l'estremità della plastica per provare a rimuoverla – se non si muove, prova con le forbici di nuovo, ma applica un po' più di pressione.



BUONE FORBICI



STARTER KIT DI ELETTRONICA



MATERIALI

CARTONE

◀ Facilmente accessibile, molto economico e semplice da usare. Il cartone non è solo ottimo per progetti più sicuri ma anche per la prototipazione di modelli o progetti 3D che usano legno e metallo

FILO DI PROTOTIPAZIONE

▶ Robusto e in grado di mantenere la sua forma, questo tipo di filo è perfetto per l'uso con la breadboard per collegare differenti componenti



CAVALLOTTI

◀ Un modo più semplice di connettere i GPIO alla breadboard è usare fili prefabbricati con connettori che si innestano facilmente sui pin GPIO



FARE MAKING COL CARTONE

Suggerimenti per costruzioni perfette

01 Misurare e segnare

Anche se il cartone è un grande materiale e potresti averne a disposizione molto, può essere una buona idea usare la carta per aiutarti a pianificare i tuoi progetti. Usa righelli e goniometri e qualsiasi altro strumento di disegno che hai, per segnare ciò che desideri fare o semplicemente tracciare la carta con cui hai sperimentato. Ci piace usare pennarelli per ottenere linee molto linea visibili, ma una matita può anche essere buona per iniziare e per farlo bene.



02 Incidere e tagliare

Per piegare il tuo cartone, dovrà incidere un po'; è quando usi una lama della forbice per correre attraverso una linea di piegatura senza tagliarlo per rendere il cartone più facile da piegare. I righelli di metallo sono ottimi per mantenere queste linee dritte. Quando tagli, taglia un po' al di fuori delle linee che hai disegnato. Puoi sempre tagliarli subito a misura, ma in caso di errore è più difficile riaggiungere il cartone.



03 Costruire

Usare delle linguette che si vanno a incassare in altre parti del cartone è un buon modo per rendere la costruzione più facile e robusta. Puoi anche usare del nastro in carta e poi passare a un nastro più resistente, usa del Blu Tack per adesioni temporanee, o piega lunghi e sottili pezzi per l'incollaggio. Scopri altri suggerimenti dai nostri amici di Adafruit: magpi.cc/cardboardtips



STRUMENTI PER RAGAZZI

Con una certa maturità arriva un po' di responsabilità. Ecco il tipo di cose a cui un adolescente potrebbe usare per fare cose incredibili e metterle sul percorso che porta ad essere un maestro maker.

STRUMENTI



SPELAFILI



SALDATORE

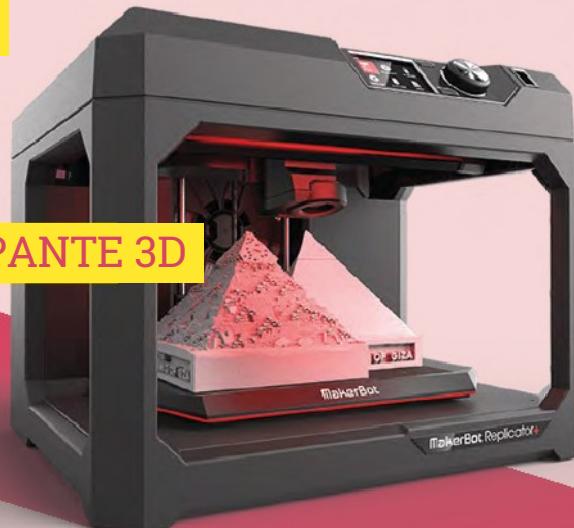
COLTELLO DA INTAGLIO



Kit di modellismo

Puoi imparare molto costruendo modellini di robot o di aerei, dall'ingegneria dei giunti a come rifinire i tagli con carta vetrata. Può anche capitare di fare vernici personalizzate. Il negozio di hobbistica più vicino a te dovrebbe averne un po', ma ci piace molto anche la Bandai Kit Gunpla.

STAMPANTE 3D



MATERIALI

STAGNO

▶ Porta le tue build a un livello successivo saldando permanentemente i circuiti utilizzando un saldatore e un po' di stagno.



CAVO ELETTRICO

▶ Questo filo intrecciato è più flessibile ma non mantiene la forma come il filo da prototipazione – questo è utile per saldare qualsiasi cosa potrebbe muoversi un po' più di una breadboard



PLA

▼ Stampa progetti 3D utilizzando una stampante 3D. PLA è solo uno dei tipi di plastica che puoi usare, anche se variano come prezzo e nell'uso



MODELLAZIONE 3D PER STAMPARE

Suggerimenti per modelli perfetti

01 Blender

Una programma di modellazione 3D gratuito, Blender (blender.org) è ottimo per creare i tuoi modelli personalizzati per la stampa 3D. Può essere però un po' complicato per i nuovi arrivati, ecco perché noi consigliamo l'eccellente Corsi di apprendimento Blender della Fondazione Raspberry Pi: magpi.cc/blenderprojects. I file finiti possono essere esportati come file .stl da dare in pasto alle stampanti 3D.



02 TinkerCAD

Un po' più semplice di Blender, TinkerCAD ti consente di creare file per la stampa 3D più semplici, molto più rapidamente di Blender – soprattutto se sei nuovo della modellazione 3D. Puoi trovarlo su tinkercad.com ed è tutto nel tuo browser. Ti consente anche di importare file immagine e simili.



03 Combining models

Un piccolo trucco che abbiamo imparato mentre giocavamo con figure in miniatura e altre cose: puoi combinare due o più modelli insieme in Blender per ulteriori modifiche. Questo è fantastico se trovi qualcosa che ti piace, ad esempio un modello di auto, e vuoi aggiungere uno spoiler più grande o un personaggio alla stampa.



STRUMENTI DA ADULTI

Puoi certamente limitarti agli strumenti e abilità delle ultime pagine per tutto il tempo che desideri, ma ecco alcuni strumenti eccellenti che scatenneranno davvero il tuo flusso creativo

STRUMENTI

TRAPANO A COLONNA



Risorse di carpenteria

Steve Ramsey su YouTube ha video eccellenti con ottimi consigli e trucchi di carpenteria, oltre a un corso online per le persone che vogliono davvero addentrarsi nella carpenteria: magpi.cc/carpentrysteve.



SALDATRICE



SEGGETTO ALTERNATIVO



PISTOLA TERMICA



MINI TRAPANO



MIGLIORAMENTI ALLO SPAZIO DI LAVORO



BANCO DA LAVORO

Imparare a saldare

La saldatura può essere uno strumento davvero utile durante il lavoro con il metallo, tuttavia può anche essere complicata e molto pericolosa. Sebbene ci siano alcuni video su YouTube, ti consigliamo di trovare un corso online o locale che puoi fare che ti insegnerà come saldare in sicurezza.



MORSA

▲ Se hai bisogno di mantenere un oggetto perfettamente fermo durante la levigatura, la sua segatura o foratura, una morsa da banco è il modo perfetto per proteggere gli oggetti quando non hai mani libere



PINZE E MORSETTI

▲ Se hai bisogno di tenere uniti tra loro degli oggetti, alcune pinze e i morsetti sono l'ideale per questo.

Spazi maker locali

La spesa per strumenti e materiali può davvero crescere parecchio, e potresti non avere lo spazio o i soldi per loro. È qui che uno spazio maker locale può essere fantastico – questi spazi comunitari tendono a riempirsi di tutta l'attrezzatura di cui potresti aver bisogno, così come le persone con esperienza da cui ricevere consigli. Potresti anche farti dei nuovi amici.

Non ci sono siti dedicati che elencano gli spazi maker, quindi dovrai usare il tuo motore di ricerca preferito, per questo.

