Raspberry Pi Desktop Kit

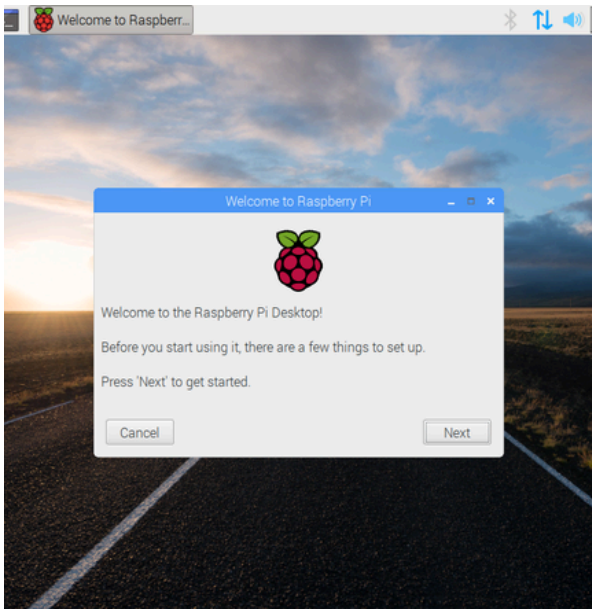
Mid-Valley
STEM-CTE HUB

www.midvalleystem.org
midvalleystemctehub@linnbenton.edu
Linn-Benton Community College
Albany Campus - CC-212

Mid-Valley
STEM-CTE HUB

Raspberry Pi Desktop Kit

The Raspberry Pi is a small computer packed with educational and imaginative potential. The Pi is designed to help you learn coding, discover how computers work, and build your own amazing things. This kit contains all of the things you will need to get started (except for the monitor) in learning how to code and build your own robotic designs. Lesson plans in the Beginner's Guide teach block coding with Scratch, traditional coding with Python, and physical computing with both.



Grade Level

4th - 12th

Group Size

1 - 4 students per set

Time Duration

60 min+ Multiple Sessions

Content of Kits

Components

- Raspberry Pi 4/5
- Mouse
- Keyboard
- Power Cable
- x2 Video Cables
- Beginner's Guide

Not Included

- Computer Monitor w/HDMI Connection (REQUIRED)

Mid-Valley
STEM-CTE HUB

Usage

Getting Started

1. The Raspberry Pi Desktop kit comes ready to plug in and get started learning how to code.

2. Beginner's Guide: Provides the user(s) with a step-by-step guide on how to introduce block coding with Scratch, coding with Python, and integrated Physical Computing with both languages to control circuits and motors.

3. Scratch Block Coding: Scratch is a high-level, block-based visual programming language and website aimed primarily at children as an educational tool, with a target audience of ages 8 to 16.[9][10] Users on the site can create projects on the website using a block-like interface. Scratch was conceived and designed through collaborative National Science Foundation grants

4. Python Coding: Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation.

Python is dynamically type-checked and garbage-collected. It supports multiple programming paradigms, including structured (particularly procedural), object-oriented, and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library.

5. Physical Computing: Physical computing is a discipline that combines hardware and software to create an interactive system. This interactive system involves the communication between the software and the physical world. It is the ability to build a tangible interactive system using hardware and software.

Storage

- Store the Raspberry Pi Desktop kit on a desktop or in its plastic container when not in use.
- Keep away from liquid or wet environments.

Troubleshooting

- For hardware or coding troubleshooting problems consult the STEM Hub or the following websites:
  - https://projects.raspberrypi.org/en/collections/scratch
  - https://www.w3schools.com/python/

Mid-Valley
STEM-CTE HUB

Activity Guide

Beginner

Programming with Scratch:
Learn how to start coding using Scratch, the block-based programming language.

Intermediate

Programming with Python:
Now you've got to grips with Scratch, learn how to do text-based coding Python.

Advanced

Physical computing with Scratch and Python:
There's more to coding than doing things on screen – you can also control electronic components connected to your Raspberry Pi's GPIO.

Extension Activities:

Create Your Own Smart Pet Feeder
Build a Smart Pet Feeder using the Raspberry Pi! Students can code in Scratch or Python to control a motor that dispenses food at set times or with the click of a button. This hands-on project combines programming, electronics, and engineering design, with opportunities to add features like sensors, LEDs, or a camera. A fun way to explore automation and physical computing!

Mid-Valley
STEM-CTE HUB

Learning Extensions

STEAM Connections: Engineering - Math - Science

Learning Objectives:

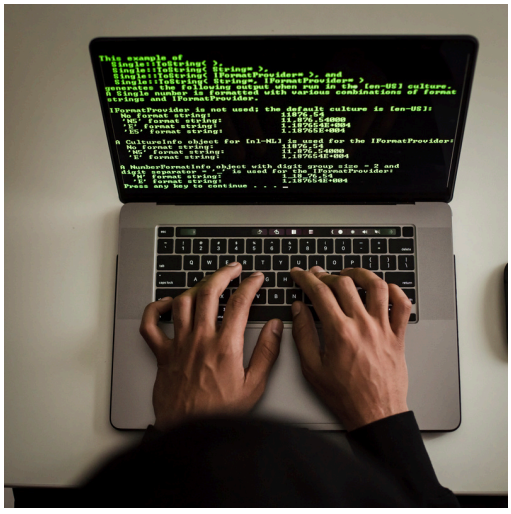
- Understand the basic components and functions of a computer using the Raspberry Pi.
- Develop foundational skills in block coding (Scratch) and text-based coding (Python).
- Apply coding knowledge to control physical components (physical computing).
- Design and build simple robotic or automated systems.
- Strengthen problem-solving and debugging skills through hands-on projects.
- Explore how software interacts with hardware to create real-world solutions.

Career Connections:

- **Software Developer / Programmer**
- **Robotics Engineer**
- **Electronics / Hardware Technician**
- **Web / App Developer**
- **IoT (Internet of Things) Developer**

Essential Employability Skills:

- **Critical thinking**
- **Problem-solving**
- **Innovation**
- **Collaboration**
- **Adaptability**
- **Communication**
- **Technical literacy**



Mid-Valley
STEM-CTE HUB

Resources and Accessibility

Safety Guidelines

- Supervise to prevent slipping accidents from spills
- Keep water and liquids away from Raspberry Pi components
- Ensure all tools are completely dry before storage to prevent mold
- Keep materials away from areas with trip hazards during active use

Accessibility

- Allow verbal responses or partner assistance for pouring steps
- Provide larger scoops or containers for students with limited motor control
- Offer extended time and clear, scaffolded steps for students needing extra support

Library Catalog

Library Resources

Feedback

QR to feedback survey