

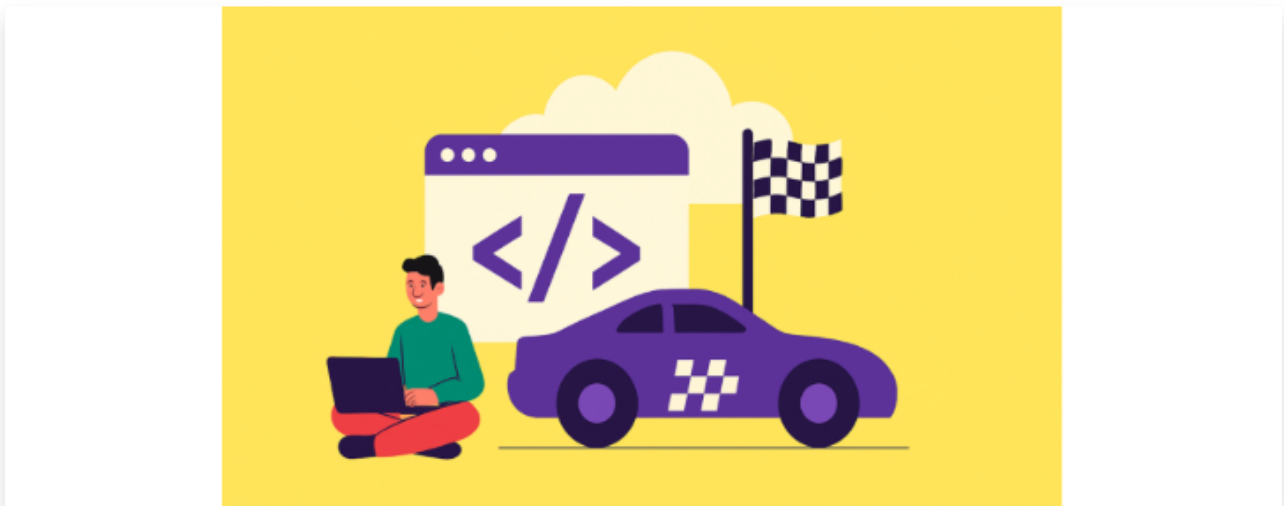
## Personal Project

---

### Textbook

---

## Personal Project



### Creating Your Racing Game

Now it's time to use everything you've learned to create your own racing game! You'll plan, build, test, and debug your own unique project.

### Project Requirements

Your racing game needs:

1. **At least 2 sprites:**
  - One or more racers (car, animal, person, or creative choice)
  - Additional sprite (another racer, obstacle, finish line, or helper)
2. **At least 2 backgrounds:**
  - Your race should cross two or more scenes
  - Examples: Start at school → race through city
  - Or: Begin in desert → finish at beach
3. **At least 5 different commands for your racing sprite:**
  - Motion blocks (direction and movement)

- Looks blocks (size changes, hide/show)
- Sound blocks (engine sounds, celebration)
- Control blocks (speed controls, loops)

4. **At least 1 loop:**

- Repeating movement
- Recurring sounds
- Multiple actions in sequence

## Planning Your Project

Before coding:

1. Choose your racing theme
2. Select sprites that fit your idea
3. Pick backgrounds for your race course
4. List the sequence of events
5. Decide what makes your race unique

## Available Resources

Remember you can use:

- **Racer sprites:** car, rocket, any animal, person sprites
- **Race backgrounds:** city, desert, beach, field, mountains
- **Special effects:** sounds, size changes, speed controls
- **Creative combinations:** flying fish, racing robots, hopping zombies

## Building Your Project

Follow these steps:

1. Start with your plan
2. Add sprites to Scene 1
3. Code the first scene
4. Test what you built
5. Add Scene 2 with Go to Page block
6. Continue building and testing

7. Debug any problems

## Success Criteria

Your project is successful when:

- All sprites move as planned
- Scene transitions work smoothly
- At least one loop functions correctly
- The race has a clear start and finish
- Your code runs without errors

## Testing Your Race

Test by checking:

- Do all sprites appear?
- Does movement work correctly?
- Do scenes change when expected?
- Does the loop repeat properly?
- Are sounds playing at right times?

## Debugging Tips

If something doesn't work:

- Check event blocks are connected
- Verify sprite selection before adding code
- Test one section at a time
- Review the order of blocks
- Make sure Go to Page blocks are placed correctly

## Project Examples

Desert Rally:

- Car sprite races through desert
- Cactus obstacles to avoid
- Speed changes for rough terrain
- Victory sound at finish

Space Race:

- Rocket racing to moon
- Star sprites as checkpoints
- Fast and slow speed sections
- Loop for engine sounds

Animal Olympics:

- Cat vs. dog race
- Multiple scene backgrounds
- Hop movements for obstacles
- Size changes for perspective

## Extending Your Project

If you finish early, add:

- More scenes to your race
- Additional racers
- Sound effects for actions
- Victory celebration sequence
- More complex loops

## Project Reflection

After completing your race:

- What worked well in your code?
- What was challenging to create?
- How did you solve problems?
- What would you add with more time?

## Critical Thinking Questions

1. How did planning help you create your project?
2. What debugging strategies did you use?
3. How did loops make your code more efficient?

## Sentence Stems

- "My project demonstrates (blank space) by (blank space)."

- "I used (blank space) blocks to create (blank space)."
- "The most challenging part was (blank space)."