## Hints for Implementation

### Hint 1: Don't Forget the Types!

Type hints make the code easier to read and debug.

### Hint 2: Planning Your Approach

Break the assignment into manageable steps:

1. **Implement the Letter class:** Focus on creating and testing this class to ensure it correctly tracks the letter’s status.
2. **Create an initial Bot class:** Start by creating a simple bot that randomly picks from the word list.
3. **Update the Bot class to make smarter guesses:** Use the feedback to adjust the bot’s logic, filtering words based on identified correct letters.
4. **Ensure the bot tracks letters not in the target word:** This will help prevent repeating incorrect guesses.
5. **Test thoroughly:** Run multiple test cases to ensure the bot adapts correctly to feedback.

### Hint 3: Considering an Algorithm for Smart Guesses

To make "smart" guesses:

- **Track correct positions:** If a letter is identified as in the correct position, restrict future guesses to maintain that position.
- **Ignore incorrect letters:** Add letters identified as incorrect to the unused_letters set to avoid them in future guesses.
- **Use filtered word lists:** The bot should narrow down possible words based on known letters and unused letters.