Implement `snake-skinny?`, which takes a snake and returns `true` if the snake weights less than 10 pounds, `false` otherwise

Design Recipe II

Data

- Understand the input data

Contract, Purpose, and Header

- Describe (but don't write) the function

Examples

- Show what will happen when the function is done

Template

- Set up the body based on the input data (and *only* the input)

Body

- The most creati

Test

- Run the exampl

Implement `snake-skinny?`, which takes a snake and returns `true` if the snake weights less than 10 pounds, `false` otherwise

Design Recipe II

Data

- Understand the input data

Contract, Purpose, and Header

- Describe (but don't write) the function

Examples

- Show what will happen when the function is done

Template

- Set up the body based on the input data (and *only* the input)

Body

- The most creati

Test

- Run the exampl

Implement `snake-skinny?`, which takes a snake and returns `true` if the snake weights less than 10 pounds, `false` otherwise

Design Recipe II

Data

- Understand the input data

Contract, Purpose, and Header

- Describe (but don't write) the function

Examples

- Show what will happen when the function is done

Template

- Set up the body based on the input data (and *only* the input)

Body

- The most creati

Test

- Run the exampl

Implement `snake-skinny?`, which takes a snake and returns `true` if the snake weights less than 10 pounds, `false` otherwise

Design Recipe II

Data

- Understand the input data

Contract, Purpose, and Header

- Describe (but don't write) the function

Examples

- Show what will happen when the function is done

Template

- Set up the body based on the input data (and *only* the input)

Body

- The most creati

Test

- Run the exampl

Implement `snake-skinny?`, which takes a snake and returns `true` if the snake weights less than 10 pounds, `false` otherwise

Design Recipe II

Data

- Understand the input data

Contract, Purpose, and Header

- Describe (but don't write) the function

Examples

- Show what will happen when the function is done

Template

- Set up the body based on the input data (and *only* the input)

Body

- The most creati

Test

- Run the exampl

Implement `snake-skinny?`, which takes a snake and returns `true` if the snake weights less than 10 pounds, `false` otherwise

Design Recipe II

Data

- Understand the input data

Contract, Purpose, and Header

- Describe (but don't write) the function

Examples

- Show what will happen when the function is done

Template

- Set up the body based on the input data (and *only* the input)

Body

- The most creati

Test

- Run the exampl

Implement `snake-skinny?`, which takes a snake and returns `true` if the snake weights less than 10 pounds, `false` otherwise

Implement **`feed-snake`**, which takes a snake and returns a snake with the same name and favorite food, but five pounds heavier

Design Recipe II

Data

- Understand the input data

Contract, Purpose, and Header

- Describe (but don't write) the function

Examples

- Show what will happen when the function is done

Template

- Set up the body based on the input data (and *only* the input)

Body

- The most creative step: implement the function body

Test

- Run the examples

Implement **`feed-snake`**, which takes a snake and returns a snake with the same name and favorite food, but five pounds heavier

Design Recipe II

Data

- Understand the input data

Contract, Purpose, and Header

- Describe (but don't write) the function

Examples

- Show what will happen when the function is done

Template

- Set up the body based on the input data (and *only* the input)

Body

- The most creative step: implement the function body

Test

- Run the examples

Implement **`feed-snake`**, which takes a snake and returns a snake with the same name and favorite food, but five pounds heavier

Design Recipe II

Data

- Understand the input data

Contract, Purpose, and Header

- Describe (but don't write) the function

Examples

- Show what will happen when the function is done

Template

- Set up the body based on the input data (and *only* the input)

Body

- The most creative step: implement the function body

Test

- Run the examples

Implement **`feed-snake`**, which takes a snake and returns a snake with the same name and favorite food, but five pounds heavier

Design Recipe II

Data

- Understand the input data

Contract, Purpose, and Header

- Describe (but don't write) the function

Examples

- Show what will happen when the function is done

Template

- Set up the body based on the input data (and *only* the input)

Body

- The most creative step: implement the function body

Test

- Run the examples

Implement **`feed-snake`**, which takes a snake and returns a snake with the same name and favorite food, but five pounds heavier

Design Recipe II

Data

- Understand the input data

Contract, Purpose, and Header

- Describe (but don't write) the function

Examples

- Show what will happen when the function is done

Template

- Set up the body based on the input data (and *only* the input)

Body

- The most creative step: implement the function body

Test

- Run the examples

Implement **`feed-snake`**, which takes a snake and returns a snake with the same name and favorite food, but five pounds heavier

Design Recipe II

Data

- Understand the input data

Contract, Purpose, and Header

- Describe (but don't write) the function

Examples

- Show what will happen when the function is done

Template

- Set up the body based on the input data (and *only* the input)

Body

- The most creative step: implement the function body

Test

- Run the examples

Implement **`feed-snake`**, which takes a snake and returns a snake with the same name and favorite food, but five pounds heavier