Object Oriented Programming

"OOP"

Introduction to Object Oriented Programming

- dot notation
  - .write()
    ```
    document.write("Hello World!");
    ```
  - .lower()
    ```
    myString = "Hello World"
    print myString.lower()
    ```

History

- Procedures - verbs
- OOP - Object Oriented Programming - nouns
- Data and Logic …encapsulated
- E.g. employee; car; player

Procedural Programming vs. OOP

Procedural

- Define tasks
- Break tasks
- Define data
- Design functions
  - input
  - output
- Group functions
- Write the code

Object Oriented Programming

- Model objects
  - Noun-oriented
  - Domain
- Phases
  - Analysis: domain
  - Design: solution implementation
  - Build it

Birth of Objects

- Models
- Cells
  - Independent, indivisible, interacting
  - Scales well
    - Complexity
    - Robustness
    - Supporting growth
    - Reusable

Concept: Classes and Objects

Classes

- attributes - properties - instance variables
- behaviors - methods - object functions
- abstract
- classes → instantiate multiple objects (object1, object2, object3, et cetera)

Encapsulation

- self-contained

Concept: Classes and Objects (cont'd)

Objects

- ```
  var myArray = [1,2,3,4,5];
  var size = myArray.length;
  document.write(size);
  ```

Primitives (not objects)

JavaScript built-in objects:

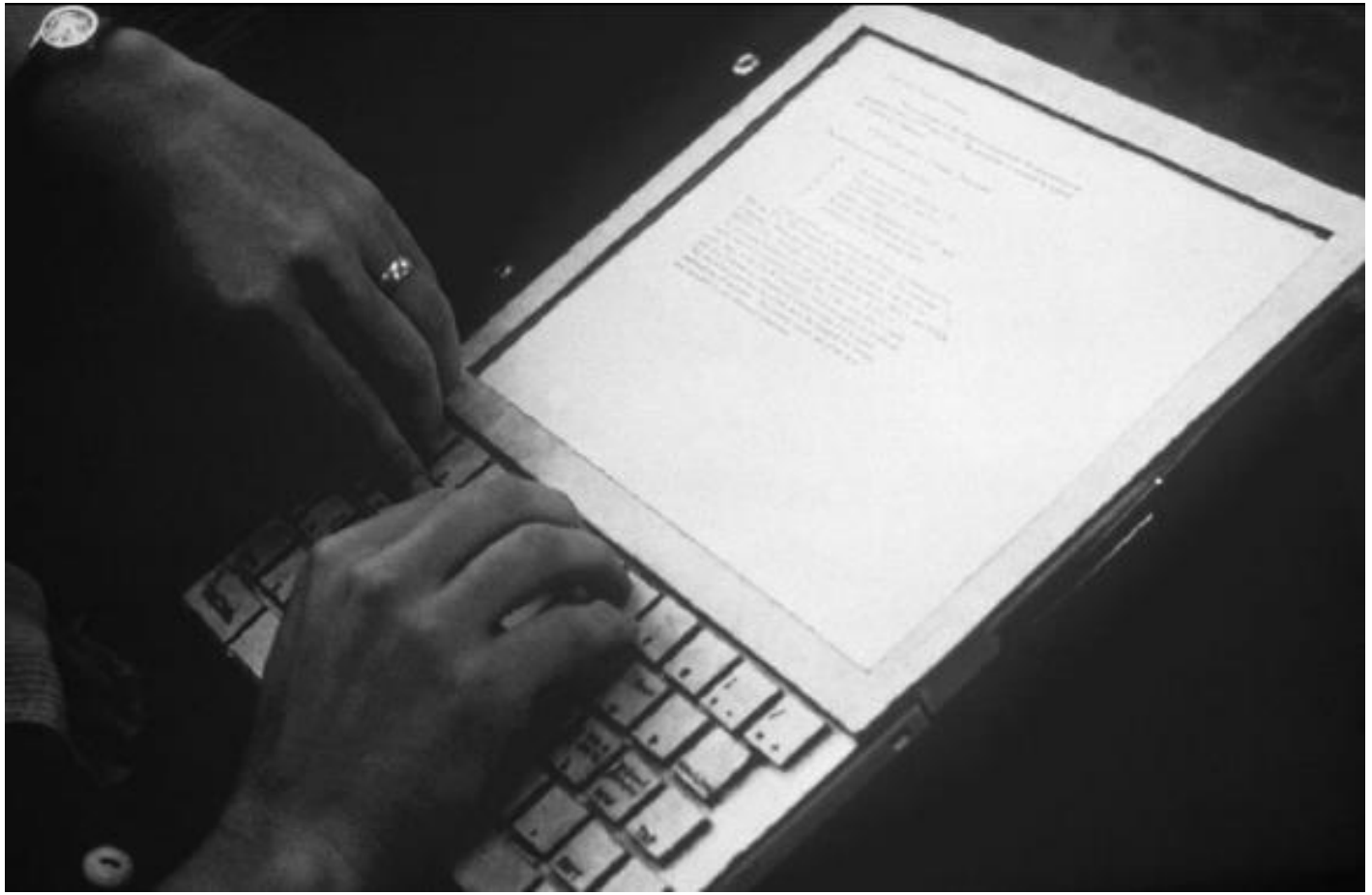
- Array
- RegExp
- Date
- Math

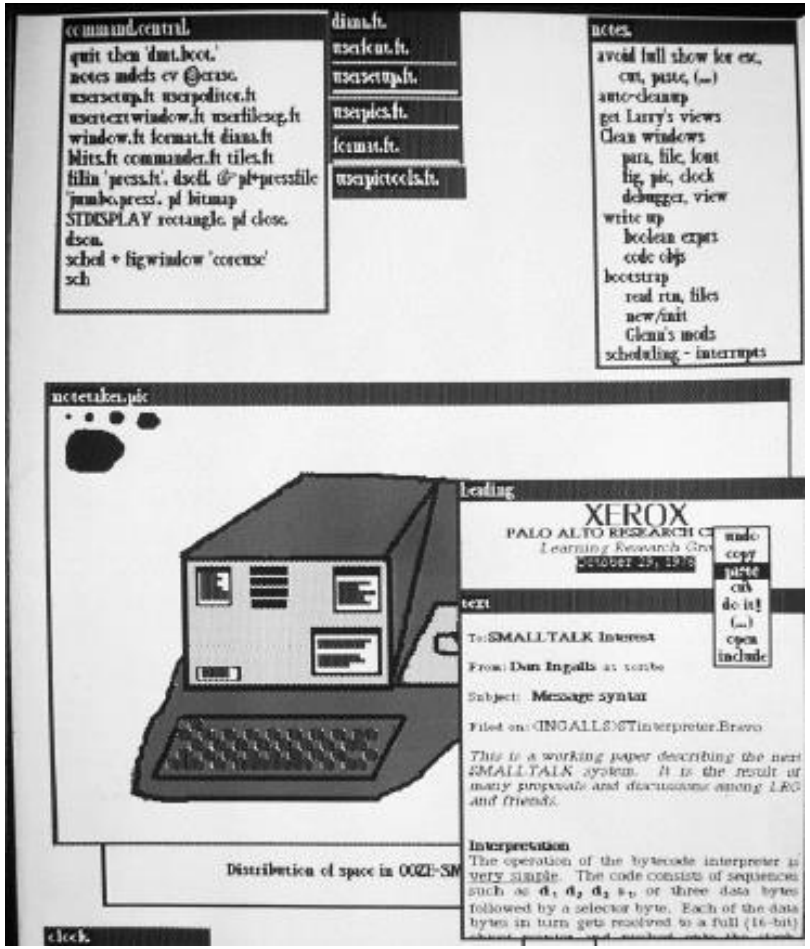
…and a few more
  ```

OOP Concepts

- Abstraction
  - Avoids duplication; increases reusability
- Encapsulation
  - Data and functions
  - Only needs to know: the interface
- Inheritance
  - Create classes and sub-classes that are related to each other
- Polymorphism
  - Ability to handle any data type
  - The "principle of least astonishment"

Prototype Dynabook
(Xerox P A R C Learning Research Group)



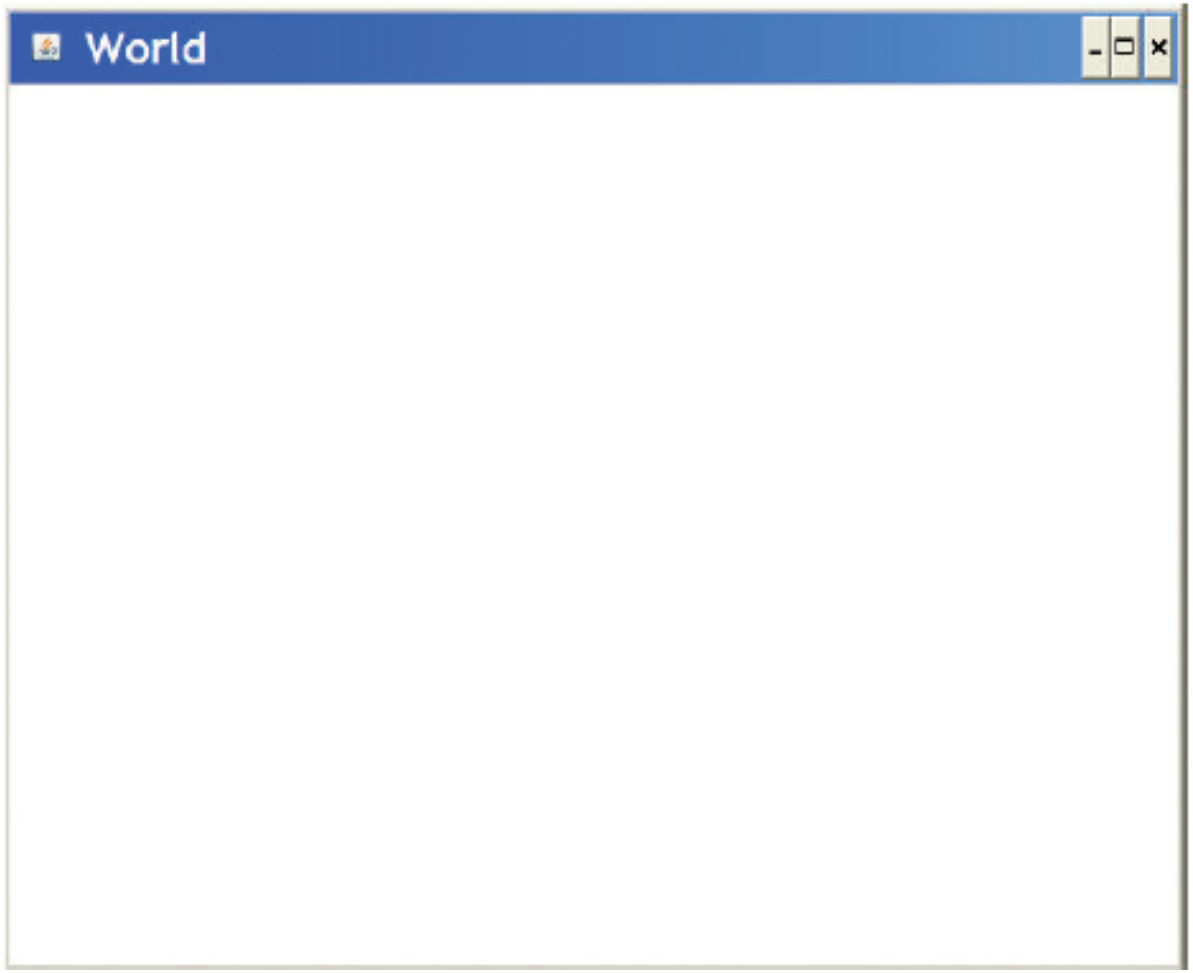


A First Object: Logo Turtle

- Dr. Seymour Papert at M I T invented the Turtle as a graphical and mathematical object to think with for the children's programming language, Logo
- A turtle is an object.
  - Every turtle understands the same methods.
  - Every turtle has the same fields or instance variables.
    - Heading, body color, pen color, X and Y position.
    - Yet each turtle can have its own values for these fields.

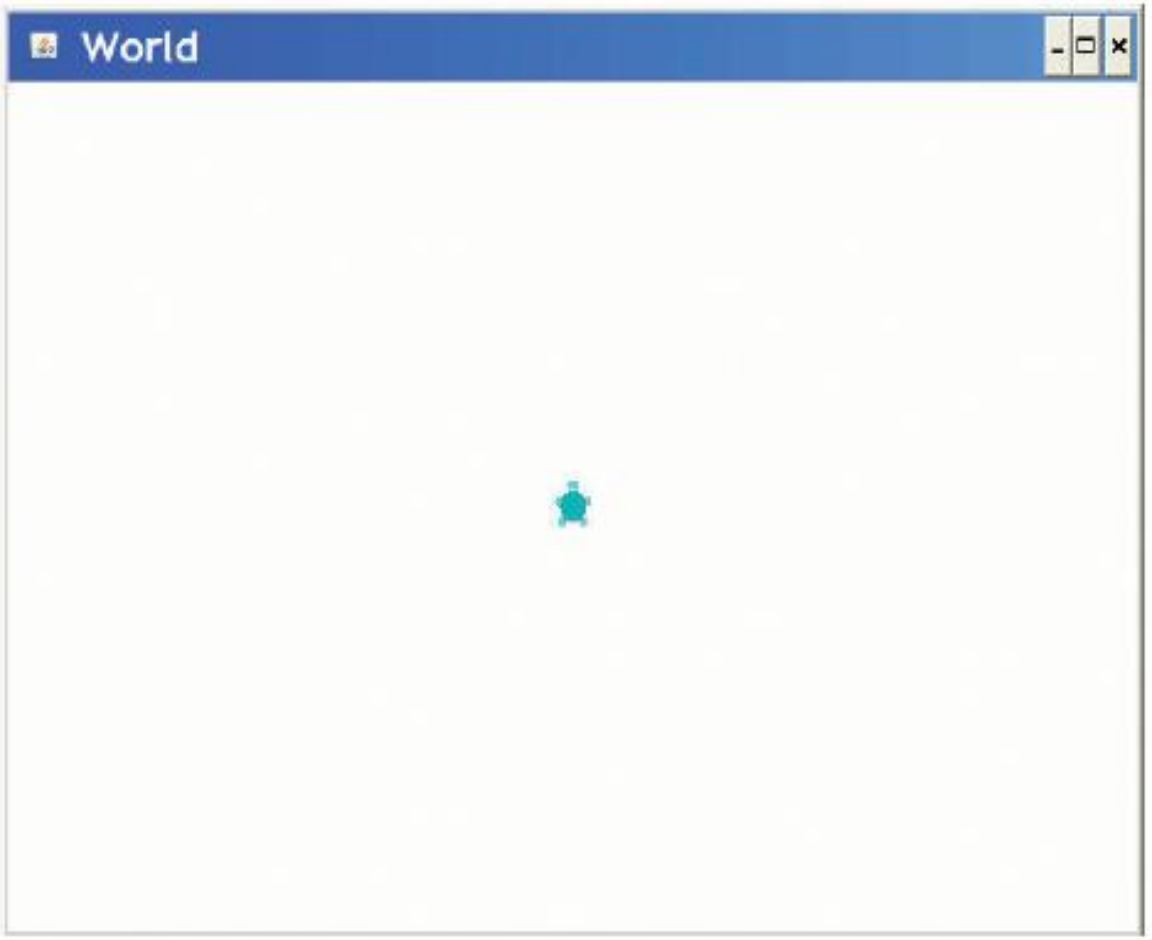
Using Turtles in Python

```
>>> makeWorld()
```



Adding a Turtle to Our World

```
>>> earth = makeWorld ()
>>> tina = makeTurtle(earth)
>>> print tina
No name turtle at 320, 240 heading 0.0.
```



Talking to Turtles as Functions or Messages/Methods

- We can tell a turtle to go forward by calling a function (telling the function to act on the turtle):

```
>>> earth = makeWorld()
>>> tina = makeTurtle(earth)
>>> forward(tina,100)
```

- Or we can ask Tina to go forward, a certain amount. We are sending a message to Tina, asking her to execute a function that only turtles know: A “method”

```
>>> earth = makeWorld()
>>> tina = makeTurtle(earth)
>>> tina.forward(100)
```

See "Methods That Turtles Understand" page 478

For Next Time

- Read Chapter 17, pages 447-464, and try sample programs

- Next time: we'll pick-up with creating *classes* in Python