

Objectives 1.1 Reasons for Studying Concepts of Programming Languages 1.2 Programming Domains
 1.3 Language Evaluation Criteria 1.4 Influences on Language Design 1.5 Language Categories 1.6
 Implementation Methods 1.8 Programming Environments Copyright (C) 2016, 2013, 2010 Pearson
 Education, Inc. All Rights Reserved Evaluation Criteria: Writability o Simplicity and orthogonality – Few
 constructs, a small number of primitives, a small set of rules for combining them o Support for
 abstraction – The ability to define and use complex structures or operations in ways that allow details to
 be ignored o Expressivity – A set of relatively convenient ways of specifying operations – Strength and
 number of operators and predefined functions Copyright (C) 2016, 2013, 2010 Pearson Education, Inc. All
 Rights Reserved Evaluation Criteria: Readability (2 of 2) o Data types – Adequate predefined data types
 o Syntax considerations – Identifier forms: flexible composition – Special words and methods of forming
 compound statements – Form and meaning: self-descriptive constructs, meaningful keywords Copyright
 (C) 2016, 2013, 2010 Pearson Education, Inc. All Rights Reserved Programming Domains (2 of 2) o
 Artificial intelligence – Symbols rather than numbers manipulated; use of linked lists – LISP o Systems
 programming – Need efficiency because of continuous use –C o Web Software – Eclectic collection of
 languages: markup (e.g., HTML), scripting (e.g., PHP), general-purpose (e.g., Java) Copyright (C) 2016,
 2013, 2010 Pearson Education, Inc. All Rights Reserved Reasons for Studying Concepts of Programming
 Languages o Increased ability to express ideas o Improved background for choosing appropriate
 languages o Increased ability to learn new languages o Better understanding of significance of
 implementation o Better use of languages that are already known o Overall advancement of computing
 Copyright (C) 2016, 2013, 2010 Pearson Education, Inc. – A manageable set of features and constructs –
 Minimal feature multiplicity – Minimal operator overloading o Orthogonality – A relatively small set of
 primitive constructs can be combined in a relatively small number of ways – Every possible combination
 is legal Copyright (C) 2016, 2013, 2010 Pearson Education, Inc. All Rights Reserved Programming
 Domains (1 of 2) o Scientific applications – Large numbers of floating point computations; use of arrays –
 Fortran o Business applications – Produce reports, use decimal numbers and characters – COBOL
 Copyright (C) 2016, 2013, 2010 Pearson Education, Inc. All Rights Reserved Language Evaluation
 Criteria o Readability: the ease with which programs can be read and understood o Writability: the ease
 with which a language can be used to create programs o Reliability: conformance to specifications (i.e.,
 performs to its specifications) o Cost: the ultimate total cost Copyright (C) 2016, 2013, 2010 Pearson
 Education, Inc. All Rights Reserved Evaluation Criteria: Readability (1 of 2) o Overall simplicity The
 .overall simplicity of a programming language strongly affects its readability