

İTÜ
DERS KATALOG FORMU
(COURSE CATALOGUE FORM)

Dersin Adı		Course Name				
Python Bilgisayar Programlamaya Giriş		Introduction to Computer Programming in Python				
Kodu (Code)	Yarıyılı (Semester)	Kredisi (Local Credits)	AKTS Kredisi (ECTS Credits)	Ders Uygulaması, Saat/Hafta (Course Implementation, Hours/Week)		
				Ders (Theoretical)	Uygulama (Tutorial)	Laboratuvar (Laboratory)
KMM 102E	2	2	5.5	1	-	2
Bölüm / Program (Department/Program)	Kimya Mühendisliği Bölümü (Department of Chemical Engineering)					
Dersin Türü (Course Type)	Zorunlu (Compulsory)			Dersin Dili (Course Language)	İngilizce (English)	
Dersin Önkoşulları (Course Prerequisites)	-					
Dersin mesleki bileşene katkısı, % (Course Category by Content, %)	Temel Bilim (Basic Sciences)	Temel Mühendislik (Engineering Science)	Mühendislik Tasarım (Engineering Design)	İnsan ve Toplum Bilim (General Education)		
	50	50				
Dersin İçeriği (Course Description)	Python'a giriş; Değişkenler ve atama ifadeleri; Fonksiyonlar; If, else ve elif komutları; Döngü yapılar; Karakter işlemler; List yapılar; Dosya işlemleri; Kimya Mühendisliği uygulamaları ile ilgili programlama alıştırmalar					
Dersin Amacı (Course Objectives)	1. Özel programlama ortamıyla yeterince aşina olunmasını sağlamak 2. Temel programlama mantığının ve tekniklerinin kavranmasını geliştirmek 3. Bir programın yazılması, derlenmesi, çalıştırılması ve hata ayıklanması bilgilerini geliştirmek 4. Bazı genel kimya mühendisliği problemlerine özgü bilgisayar algoritmaları ve programlama dili üzerine işleyiş bilgisi geliştirmek 5. İleri düzey programlama dili kullanarak öğrencilerin problemleri analiz etme ve çözme kabiliyetini geliştirmeye odaklanmak					
Dersin Öğrenme Çıktıları (Course Learning Outcomes)	1. to develop enough familiarity with the specific programming environment 2. to develop an understanding of fundamental programming logic and programming techniques 3. to develop the knowledge of editing, compiling, running and debugging of a program 4. to develop a working knowledge on the computer algorithms and programming language specific to some common chemical engineering problems 5. to emphasize on developing the students' ability to analyze and solve problems by using high level programming language					
Dersin Öğrenme Çıktıları (Course Learning Outcomes)	Öğrenciler 1. programlama dilinin sözdizimini ve yapısını anlayabilecek 2. bir problemi analiz edebilecek ve algoritma geliştirebilecek 3. bir programı test edebilecek, derleyebilecek, hata ayıklayabilecek ve doğrulama yapabilecek 4. prosedürel, prosedürel olmayan, mantık, fonksiyonel pratik programlama becerilerini geliştirecek 5. kapsamlı örneklerin gereksinimlerini karşılamak için bir program tasarlayabilecek 6. düzgün bir programlama dökümanı ve rapor yazabilecek					
Dersin Öğrenme Çıktıları (Course Learning Outcomes)	Students will be able to, 1. understand the syntax and structure of the programming language 2. analyze a problem and develop an algorithm 3. test, compile, debug, and verify the program 4. develop practical programming skills in procedural, nonprocedural, logic, functional 5. design a program to meet requirements of comprehensive examples 6. write appropriate program documentation and report					

Ders Kitabı (Textbook)	A Primer on Scientific Programming with Python 5 th Edition, Hans Petter Langtangen, Springer 2016.		
Diğer Kaynaklar (Other References)	<ol style="list-style-type: none"> 1. Python Programming Fundamentals, Kent D. Lee, Springer-Verlag 2014 2. Learn Python Programming : A Beginner's Guide to Learning the Fundamentals of Python Language to Write Efficient, High-Quality Code, 2nd Edition, Fabrizio Romano, Packt Publishing, Limited 2018 3. An Introduction to Python and Computer Programming, Yue Zhang, Springer Science+Business Media 2015 4. Fundamentals of Python Programming, R. L. Halterman, 2016 5. Learn to Program with Python, I. Kalb 2016 		
Ödevler ve Projeler (Homework & Projects)	5 ödev (en az)		
	5 homework assignment (at least)		
Laboratuvar Uygulamaları (Laboratory Work)	Derse eşgereklilik olarak ilgili laboratuvar oturumu vardır.		
	There is an associated laboratory session which is a co-requisite.		
Bilgisayar Kullanımı (Computer Use)	Bu derste bilgisayar kullanımı zorunludur.		
	Computer use in this course is compulsory		
Diğer Uygulamalar (Other Activities)	-		
	-		
Başarı Değerlendirme Sistemi (Assessment Criteria)	Faaliyetler (Activities)	Adedi (Quantity)	Değerlendirmedeki Katkısı, % (Effects on Grading, %)
	Yıl İçi Sınavları (Midterm Exams)	1	30
	Kısa Sınavlar (Quizzes)	-	-
	Ödevler (Homework)	5	30
	Projeler (Projects)	-	-
	Dönem Ödevi/Projesi (Term Paper/Project)	-	-
	Laboratuvar Uygulaması (Laboratory Work)	-	-
	Diğer Uygulamalar (Other Activities)	-	-
	Final Sınavı (Final Exam)	1	40

DERS PLANI

Hafta	Konular	Dersin Çıktıları
1	Python'a Giriş: Python nedir?; Python kurulumu; Python ile yazılmış bir bilgisayar programının irdelenmesi (bir oyun programı); Python ile ilk program yazımı; Bir Python programının oluşturulması, saklanması ve çalıştırılması	1
2	Değişkenler ve Atama İfadeleri: Programlama blok yapısının oluşturulması; Veri tipleri: tam sayı, gerçek sayı, karakter, boolean (mantık); Değişkenler; Değişken isimleri; Python'da korunmuş kelimeler; atama ifadeleri; Değişken atama yapıları; Aritmetik işlemler ve öncelik; İşlem sıralaması; Açıklama ekleme	1, 2, 3
3	Yazım ve sayıların formatlanması; Print komutu; Hatalar ve hatalardan kaçınma; Bilgisayar programlama stiline giriş: analiz, algoritma tasarım, program kod yazımı, test etme; Bir programın yalancı kod halinde yazılması: akış diyagramları	1, 2, 3
4	Fonksiyonlar: Matematik fonksiyonlar; Fonksiyon çağırma; Argumanlar; Hazır fonksiyon; Kullanıcıdan girdi sağlanması; Dönüştürme fonksiyonları (int, float, str); Kullanıcı tanımlı fonksiyonlar; Bir fonksiyon tanımlama; Kullanıcı tanımlı bir fonksiyonun yapılandırılması; Kullanıcı tanımlı bir fonksiyonun çağırılması; Bir fonksiyondan değer gönderimi (return komutu); Verilerin ana programa gelmesi: değer olmadan, birden fazla değere sahip olarak; Çağırma ve fonksiyon işlemlerinde özel ve genel değişken isimleri; Python dosyasına fonksiyon yerleştirme; Sabitler; Fonksiyonda hata belirleme	1, 4, 6
5	If, else ve elif komutları: Akış diyagramları; If komutu; Karşılaştırma operatörleri; if-else komut yapısı; İç içe if yapısı; elif komutu; Örnek program (bir not oluşturma programı); Koşullu mantık – koşul operatörleri; Mantık oreratörleri (not, and, or);	1, 2, 3, 4, 6
6	Karşılaştırma ve mantık operatörlerin önceliği; Koşullu ifadeler; Bir fonksiyonda if/else kullanımı; Yaygın yapılan hatalar; Uygulama	1, 2, 3, 4, 6
7	Döngüler: Döngü yapılar; while komutu; Artım ve azalım; Bir programın sonlanmadan defalarca çalıştırılması; Python hazır paketler; Rastgele bir sayının oluşturulması; Yazı tura simülasyonu; Sonsuz bir döngü yaratılması; Uygulama	3, 4, 6
8	for döngüsü; range komutu; İç içe yapılar; while true, break ve continue komutları; Kullanıcıya tekrar işlem yapıp yapmayacağını sorulması; "sayı tahmin" programı; try/except yapısı ile hata kontrolü; Uygulama	3, 4, 6
9	Karakter işlemler: Karakter yapıları ait fonksiyonlar; Bir karakter veri tipinde indis olayı; Dilimleme operatörü; (+) ekleme ve (*) tekrar etme operatörleri; in ve not in operatörleri; Karakter yapıların karşılaştırılması; Bir karakter veri yapısındaki karakterlere sıralı erişim; Hazır karakter işlemler; Karakter tipine uygulanan yaygın kullanılan operasyonlar; Karakter modüller; Uygulama	1, 3
10	List yapı: Veri koleksiyonları; list yapı; Öğeler; Boş liste; Bir list yapıdaki bir öğenin konumu (indis); Bir list yapıdaki bir öğeye erişim; Bir list yapıda indis olarak değişken veya bir ifade kullanımı; Bir list yapıdaki değerlerin değiştirilmesi; Negatif indis kullanımı; Bir list yapıdaki elemanların sayısının belirlenmesi (len fonksiyonu);	2, 3, 4
11	List yapılar için fonksiyonlar; List yapılarda +, * ve in / not in operatörleri; List yapıların manipülasyonu (list metodlar); Bir listenin tüm elemanlarına erişim (for döngüsü); List yapı kavramalar; Uygulama	2, 3, 4
12	İki boyutlu list yapılar (iç içe list yapılar); List yapılar ile ilgili daha fazla işlemler; Tuple yapısı; Uygulama	2, 3, 4
13	Dosya Girdi/Çıktı İşlemleri: Bir dosyanın açılması; Bir dosyaya veri yazılması; Bir dosyadan veri okunması; Bir dosyadan tüm verilerin okunması; Sayısal verilerin yazılması ve okunması; Uygulama	4, 6
14	Kimya Mühendisliği uygulamaları ile ilgili programlama alıştırmalar	3, 4, 5

COURSE PLAN

Weeks	Topics	Course Outcomes
1	Introduction to Python: What is Python?; Installing Python; Analyzing a computer program written in Python (a game program); Writing first program in Python; Creating, Saving and Running a Python program	1
2	Variables and Assignment Statements: The building blocks of programming; Data types: Integers, Floats, Strings, Booleans; Variables; Variable names; Reserved words in Python; Assignment statement; Arithmetic operators and precedence, Order of operations; Adding comments	1, 2, 3
3	Formatting text and numbers; Print statement; Errors and avoiding errors; Introduction to computer programming style : analysis, algorithm design, program coding, testing; Writing a program in pseudocode: Flowing charts; Exercises	1, 2, 3
4	Functions : Math functions; Function call; Arguments; Built-in type function; Getting input from the user; conversion functions (int, float, str); User Defined Functions: Definition of a function, Building a user-defined function; Calling a user-defined function; Returning a value from a function (return statement); Returning values : no value, more than one value; Specific and General Variable Names in Calls and Functions; Placement of Functions in a Python file; Constants; Find errors in functions	1, 4, 6
5	If, else, and elif statements: Flowcharting; The if statement; Comparison operators; if - else statement ; Nested if statement; The elif statement; Example Program (a grading program); Conditional logic – conditional operators; The logical operators (not, and, or)	1, 2, 3, 4, 6
6	Precedence of comparison and logical operators; Inline if usage (conditional expressions); Using if/else inside a function; Common mistakes; Exercises	1, 2, 3, 4, 6
7	Loops : Loops; The while statement; Increment and decrement; Running a program multiple times; Python Built-in packages; Generating a random number; Simulation of flipping a coin; creating an infinite loop; Exercises	3, 4, 6
8	for loops; range statement; nested structures; ile true, break, and continue statements; Asking if the user wants to repeat; “guess the number” program; error checking with try/except ; Exercises	3, 4, 6
9	Strings : Functions for strings; Indexing characters in a string; The slicing operator; The concatenation (+) and repetition (*) operators; <i>in</i> and <i>not in</i> operators; Comparing strings; Iterating characters in a string; built-in string operations; Common operations on strings; String module; Examples	1, 3
10	Lists : Collections of data; lists; elements; empty list; position of an element in a list (index); accessing an element in a list; using a variable or expression as an index in a list; changing values in a list; using negative indices; determining the number of elements in a list (the len function)	2, 3, 4
11	Functions for lists; +, *, and <i>in / not in</i> operators for lists; List manipulation (list methods); Traversing elements in a for loop; List comprehensions; Exercises	2, 3, 4
12	Two-dimensional lists (Nested lists); More with lists; Tuples ; Exercises	2, 3, 4
13	File Input/Output : Opening a file; Writing data to a file; Reading data from a file; Reading all data from a file; Writing and reading numeric data ; Exercises	4, 6
14	Programming exercises related with chemical engineering applications	3, 4, 5

Dersin Kimya Mühendisliği Programıyla İlişkisi

	Program Çıktıları	Katkı Seviyesi		
		1	2	3
b	Kimya Mühendisliği ve ilgili alanlardaki mühendislik problemlerini saptama, tanımlama ve çözme becerisi	x		
f	Modern mühendislik teknik ve araçları ile bilişim teknolojilerini etkin bir şekilde kullanma becerisi			x
h	Bireysel çalışma becerisi		x	

1: Düşük, 2. Orta, 3. Yüksek

Relationship between the Course and Chemical Engineering Curriculum

	Program Outcomes	Level of Contribution		
		1	2	3
b	An ability to identify, formulate, and solve engineering problems in chemical engineering and related fields	x		
f	An ability to use the techniques, skills, and modern engineering and computing tools necessary for engineering practice			x
h	An ability to function independently		x	

1: Low, 2. Moderate, 3. High

<u>Düzenleyen (Prepared by)</u> Prof.Dr. Reha Yavuz	<u>Tarih (Date)</u> 27.02.2021	<u>İmza (Signature)</u>
--	-----------------------------------	-------------------------