

Otter (teachR)

Selbstlernutorials für Frankfurter Studierende, um R-Grundlagen zu lernen.

Das Tutorial kann seminarbegleitend oder unabhängig von der Lehre selbstständig von Studierenden eingesetzt werden.

Planned ECTS: , **Number of learners:** 300, **Mode of delivery:** Online

Status: COMPLETED, **Course public access:** Public

Contributors:Meike Steinhilber, Patricia Heise

Course learning outcome	Level	Weight
Die Studierenden können verschiedene Datentypen unterscheiden	Remembering	10
Die Studierenden verstehen die R-Syntax	Understanding	20
Die Studierenden können selbstständig Fehler im Code beheben	Analysing	20
Die Studierenden können eigenständig Code verfassen	Applying	20
Die Studierenden können ihre eigenen Daten an vorliegenden Code anpassen	Applying	10
Die Studierenden können vorgegebenen (fremden) Code anpassen	Applying	20
		Total Weight: 100

Topic / Unit name	Workload	Learning type	Mode of delivery			Groups	Collaboration	Feedback	Assessment		
									Points	Types	Providers
R & RStudio Grundlagen											
Oberfläche bedienen können											
Screenshots anschauen und Beschreibung lesen	5 min	Practice	Online	Asynchronous	Teacher not present	No	No	Automated	No		
Total unit workload	0.08h										
Übungen zu RStudio machen											
Übung zu RStudio	10 min	Practice	Online	Asynchronous	Teacher not present	No	No	No	No		
Total unit workload	0.16h										

Topic / Unit name	Workload	Learning type	Mode of delivery	Groups	Collaboration	Feedback	Assessment		
							Points	Types	Providers
<h2>Datentypen</h2> <p>Die Studierenden verstehen die R-Syntax (35%), Die Studierenden können eigenständig Code verfassen (10%), Die Studierenden können ihre eigenen Daten an vorliegenden Code anpassen (10%), Die Studierenden können selbstständig Fehler im Code beheben (10%)</p>									
<h2>Logische Operatoren</h2>									
Logische Operatoren kennenlernen Übersicht und Information über logische Operatoren geben. Mit Beispielcode an einem Praxisbeispiel Funktion aufzeigen.	5 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No
Logische Operatoren üben Übungsaufgaben, in denen die Lernenden logische Operatoren anwenden müssen.	15 min	Practice	Online	Asynchronous	Teacher not present	No	No	Automated	No
Total unit workload	0.33h								
<h2>Integer</h2>									

<p>Logische Operatoren kennenlernen Integer erklären. Mit Beispielcode an einem Praxisbeispiel Funktion aufzeigen.</p> <ul style="list-style-type: none"> • Erstellen von integer • Klasse prüfen • von doubles abgrenzen können 	5 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No	
Total unit workload	0.08h									
Double										
<p>Double kennenlernen Double kennenlernen</p> <ul style="list-style-type: none"> • Grundlagen erklären • Einsatzmöglichkeiten • Klasse überprüfen • fehlende Werte (`NaN`) einführen • Sonderfälle erklären 	5 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No	
<p>Double üben Übungsaufgaben, in denen Lernende mit Doubles arbeiten müssen.</p>	15 min	Practice	Online	Asynchronous	Teacher not present	No	No	Automated	No	
Total unit workload	0.33h									
Character										

Character kennenlernen Double kennenlernen <ul style="list-style-type: none"> • Grundlagen erklären (Nutzung von Anführungszeichen) • Einsatzmöglichkeiten (als Daten (Text) oder auch in R (Farben für Grafiken etc.)) • Klasse überprüfen • erste Packages einfließen lassen 	15 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No
Character üben Übungsaufgaben, in denen Lernende mit Characters arbeiten müssen. <ul style="list-style-type: none"> • Code komplett neuschreiben • Code anpassen 	10 min	Practice	Online	Asynchronous	Teacher not present	No	No	Automated	No
Total unit workload	0.41h								
Fehlende Werte									

<p>Fehlende Werte kennenlernen Verschiedene fehlende Werte kennenlernen</p> <ul style="list-style-type: none"> • NaN • NA • NULL • logical(0) • integer(0) • double(0) <p>Allgemeines zu fehlenden Werten:</p> <ul style="list-style-type: none"> • Länge des Objekts überprüfen • Fehlende Werte in Rechenoperationen • auf fehlende Werte prüfen <p>Beispielcode!</p>	20 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No
<p>Umgang mit fehlenden Werten üben Übungsaufgaben, in denen Lernende mit fehlenden Werten arbeiten müssen.</p> <ul style="list-style-type: none"> • Fehler im Code korrigieren • Multiple Choice zu allgemeinen (logischen) Fragen zu fehlenden Werten 	15 min	Practice	Online	Asynchronous	Teacher not present	No	No	Automated	No
<p>Total unit workload</p>	0.58h								

Topic / Unit name	Workload	Learning type	Mode of delivery	Groups	Collaboration	Feedback	Assessment		
							Points	Types	Providers
<h2>Datenstrukturen</h2> <p>Die Studierenden können verschiedene Datentypen unterscheiden (20%), Die Studierenden verstehen die R-Syntax (10%), Die Studierenden können eigenständig Code verfassen (20%), Die Studierenden können ihre eigenen Daten an vorliegenden Code anpassen (10%), Die Studierenden können vorgegebenen (fremden) Code anpassen (20%), Die Studierenden können selbstständig Fehler im Code beheben (20%)</p>									
<h2>Variablen</h2>									
Variablen kennenlernen Übersicht und Information über Variablen geben. Mit Beispielcode an einem Praxisbeispiel Funktion aufzeigen, Mit Screenshot Aussehen in Umgebung erklären.	15 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No
Variable erstellen Variable mit vorgegebenen Werten erstellen.	15 min	Production	Online	Asynchronous	Teacher not present	No	No	Automated	No
Total unit workload	0.5h								
<h2>Atomare Vektoren</h2>									

Atomare Vektoren erklären Atomare Vektoren erklären <ul style="list-style-type: none"> • Funktion <code>c()</code> vorstellen • Klammern <code>[]</code> einführen • Unterschied Skalare und klassische Vektoren Beispielcodes: <ul style="list-style-type: none"> • Ansteuern mit Klammern zeigen • Ansteuern mit <code>TRUE</code> und <code>FALSE</code> zeigen • Vektorisierte Berechnungen 	10 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No
Mit Variablen arbeiten Übungen zu verschiedenen Unterthemen <ul style="list-style-type: none"> • Variablen erstellen • einzelne Werte in Variablen ersetzen • mit Vektoren rechnen 	20 min	Production	Online	Asynchronous	Teacher not present	No	No	Automated	No
Total unit workload	0.5h								
Faktoren									

Atomare Vektoren erklären Faktoren erklären <ul style="list-style-type: none"> als kategoriale Variablen Beispiele geben (z.B. Geschlecht) Beispielcode	15 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No
Factors erstellen Übungsaufgabe, um mit Factors zu arbeiten <ul style="list-style-type: none"> Factor erstellen Factor sortieren 	15 min	Practice	Online	Asynchronous	Teacher not present	No	No	Automated	No
Total unit workload	0.5h								
Matrix									
Matrix erklären Matrix einführen <ul style="list-style-type: none"> Beschränkungen aufführen (Class) Klammern [] und Orientierung erklären 	15 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No
Matrix erstellen Matrixes selbst erstellen	5 min	Production	Online	Asynchronous	Teacher not present	No	No	Automated	No
Matrix organisieren Matixes ansteuern <ul style="list-style-type: none"> Spalten Zeilen 	5 min	Practice	Online	Asynchronous	Teacher not present	No	No	Automated	No
Total unit workload	0.41h								

Topic / Unit name	Workload	Learning type	Mode of delivery	Groups	Collaboration	Feedback	Assessment		
							Points	Types	Providers
<h2>Funktionen</h2> <p>Die Studierenden verstehen die R-Syntax (50%), Die Studierenden können eigenständig Code verfassen (50%)</p>									
<h3>Funktionen erstellen</h3>									
Funktionen erklären Funktionen erklären: <ul style="list-style-type: none"> • Struktur • Syntax Beispielcode zeigen	15 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	Automated	No
Einfache Funktionen erstellen Kleine Funktionen zu vorgegebenen Aufgaben erstellen (z.B. Funktion, die zwei Zahlen addiert)	5 min	Production	Online	Asynchronous	Teacher not present	No	No	No	No
Total unit workload	0.33h								

Topic / Unit name	Workload	Learning type	Mode of delivery	Groups	Collaboration	Feedback	Assessment		
							Points	Types	Providers
<h2>Datenverarbeitung</h2> <p>Die Studierenden können verschiedene Datentypen unterscheiden (20%), Die Studierenden können ihre eigenen Daten an vorliegenden Code anpassen (70%), Die Studierenden können vorgegebenen (fremden) Code anpassen (10%)</p>									
<h3>Datenverarbeitung</h3>									
Beispieldatensatz zeigen	5 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No
Daten verarbeiten <ul style="list-style-type: none"> • Daten anzeigen (bestimmte Werte) • erste deskriptiven Statistiken • fehlende Werte anzeigen lassen • fehlende Werte entfernen • Daten ändern 	20 min	Practice	Online	Asynchronous	Teacher not present	No	No	Automated	No
Total unit workload	0.41h								
<h3>Daten importieren/exportieren</h3>									
Speichertypen erklären (Export) Verschiedene Speicherformate einführen und Vor- und Nachteile erklären (mit Beispielcode!) <ul style="list-style-type: none"> • .csv • .rda oder RData • .rds 	10 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No

Daten importieren Beispielcode zum importieren verschiedener Datentypen	10 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No
Total unit workload	0.33h								

Topic / Unit name	Workload	Learning type	Mode of delivery		Groups	Collaboration	Feedback	Assessment		
								Points	Types	Providers
Code Flow										
Die Studierenden verstehen die R-Syntax (30%), Die Studierenden können eigenständig Code verfassen (30%), Die Studierenden können vorgegebenen (fremden) Code anpassen (30%), Die Studierenden können selbstständig Fehler im Code beheben (10%)										
Einführung/Erklärung in Code Flow										
Code Flow Einführung Problem erklären und Lösungsvorschläge geben • if-Statements • for Loops • while Loops	15 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No	No
Total unit workload	0.25h									
If-else										
If-else-Statements erklären Erklärung zu verschiedenen Conditions geben Beispielcodes zeigen Auch Verknüpfung verschiedener Loops	15 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No	No
If-else-Statements üben Code mit if-else selbst produzieren	20 min	Production	Online	Asynchronous	Teacher not present	No	No	Automated	No	No

If-else-Statements reflektieren If-else vorgeben und über Output nachdenken lassen	15 min	Investigation	Online	Asynchronous	Teacher not present	No	No	Automated	No
Total unit workload	0.83h								
For-Loop									
For-Loops erklären Beispielcodes (Beispiel mit length() zeigen)	15 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No
For-Loop erstellen Code mit For-Loop selbst produzieren	15 min	Production	Online	Asynchronous	Teacher not present	No	No	Automated	No
Total unit workload	0.5h								
While-Loop									
While-Loops erklären Beispielcodes	15 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No
While-Loop erstellen Code mit While-Loop selbst produzieren	15 min	Production	Online	Asynchronous	Teacher not present	No	No	Automated	No
Negativbeispiel bearbeiten Negativbeispiel aufzeigen	10 min	Investigation	Online	Asynchronous	Teacher present	No	No	No	No
Total unit workload	0.66h								

Topic / Unit name	Workload	Learning type	Mode of delivery	Groups	Collaboration	Feedback	Assessment		
							Points	Types	Providers
<h2>Debugging</h2> <p>Die Studierenden verstehen die R-Syntax (30%), Die Studierenden können eigenständig Code verfassen (30%), Die Studierenden können vorgegebenen (fremden) Code anpassen (10%), Die Studierenden können selbstständig Fehler im Code beheben (30%)</p>									
<h3>Warnungen und Fehler</h3>									
<p>Warnungen und Fehler unterscheiden Warnings/Errors einführen</p> <ul style="list-style-type: none"> • Unterschied Warning/Error • Funktionsweise Error (if-Statement) • eigene Funktionen und Warnungen <p>Beispielcode zeigen</p> <ul style="list-style-type: none"> • Warnung selbst schreiben • Error selbst schreiben 	15 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	Automated	No
<p>Warnung selbst schreiben Übungsaufgabe, wo Studierende selbst eine simple Warnung schreiben sollen</p>	15 min	Production	Online	Asynchronous	Teacher not present	No	No	Automated	No

<p>Tipps Weiterführende Informationen und Tipps rund um Errors/Warnings</p> <ul style="list-style-type: none"> • Help-Seite von Funktionen • häufige Fehler vermeiden • etc. 	15 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	Automated	No
<p>Total unit workload</p>	0.75h								

Topic / Unit name	Workload	Learning type	Mode of delivery			Groups	Collaboration	Feedback	Assessment		
									Points	Types	Providers
Umgebung(en)											
Die Studierenden verstehen die R-Syntax (100%)											
Funktionen											
Funktionen	20 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No		
Loops	10 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No		
Total unit workload	0.5h										

Topic / Unit name	Workload	Learning type	Mode of delivery	Groups	Collaboration	Feedback	Assessment				
							Points	Types	Providers		
Übergeordnete Übungen											
Die Studierenden können eigenständig Code verfassen (7%), Die Studierenden können ihre eigenen Daten an vorliegenden Code anpassen (10%), Die Studierenden können vorgegebenen (fremden) Code anpassen (10%), Die Studierenden können selbstständig Fehler im Code beheben (10%)											
Debugging											
Fehler beheben Zunächst einfache, dann schwerere Übungen mit vorgebenen Fehlern	30 min	Investigation	Online	Asynchronous	Teacher not present	No	No	Automated	0	Summative	Automated
Total unit workload	0.5h										
Code umschreiben											
Übung Code umschreiben Vorgegebener (funktionierender) Code, der umgeschrieben werden soll, ohne Funktionalität zu verlieren	20 min	Production	Online	Asynchronous	Teacher not present	No	No	Automated	No		
Total unit workload	0.33h										
Datensätze nutzen											

Erstellung eines Datensatzes Code zeigen, der Erstellung eines Datensatzes zeigt	10 min	Acquisition	Online	Asynchronous	Teacher not present	No	No	No	No
Datensatz-Navigation üben Übungsaufgaben, in denen Studierende lernen sollen, mit einem vorgegebenen Datensatz umzugehen <ul style="list-style-type: none"> • Fehlende Werte • bestimmte Zeilen ausgeben lassen 	20 min	Production	Online	Asynchronous	Teacher not present	No	No	Automated	No
Total unit workload	0.5h								
Komplexe Programmierübungen									
allgemeine Programmierübungen	120 min	Production	Online	Asynchronous	Teacher not present	No	No	Automated	No
Total unit workload	2h								

Topic / Unit name	Workload	Learning type	Mode of delivery			Groups	Collaboration	Feedback	Assessment		
									Points	Types	Providers
<h2>Freies Üben</h2> <p>Die Studierenden können eigenständig Code verfassen (100%)</p>											
Code Chunk											
Code Chunk Leerer Codechunk (ähnlich in RStudio), wo frei gecodet werden kann	240 min	Production	Online	Asynchronous	Teacher not present	No	No	Automated	No		
Total unit workload	4h										

Total course workload: 16.33h