Otter (teachR)

Otter (teachR)

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

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

Arbeitsbelastung in Stunden: 15

Anzahl der Lernenden: 300

Bereitstellungsmodus: Online

Status: ABGESCHLOSSEN

Öffentlicher Zugang zum Kurs: Öffentlich

Mitwirkende:
Meike Steinhilber, Patricia Heise

Kurs-Lernergebnis	Niveau	Gewicht
Die Studierenden können verschiedene Datentypen unterscheiden	Erinnern	10
Die Studierenden verstehen die R-Syntax	Verstehen	20
Die Studierenden können selbstständig Fehler im Code beheben	Analysieren	20
Die Studierenden können eigenständig Code verfassen	Anwenden	20
Die Studierenden können ihre eigenen Daten an vorliegenden Code anpassen	Anwenden	10
Die Studierenden können vorgegebenen (fremden) Code anpassen	Anwenden	20

Gesamtgewicht: 100

Thema / Einheitsname	Arbeitsbelastung	Lernart	Lieferart			Gruppen	Zusammenarbeit	Rückmeldungen	Verpflichtende Aktivität	Bewertung		
										Punkte	Typ	Anbieter
R & RStudio Grudlagen												
Oberfläche bedienen können												
Screenshots anschauen und Beschreibung lesen	5 min	Übung	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein		
Gesamtarbeitsbelastung der Einheit	0.08h											
Übungen zu RStudio machen												

Übung zu RStudio	10 min	Übung	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
Gesamtarbeitsbelastung der Einheit	0.16h									
Datentypen 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%)**										
Logische Operatoren										
Logische Operatoren kennenlernen Übersicht und Information über logische Operatoren geben. Mit Beispielcode an einem Praxisbeispiel Funktion aufzeigen.	5 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
Logische Operatoren üben Übungsaufgaben, in denen die Lernenden logische Operatoren anwenden müssen.	15 min	Übung	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein
Gesamtarbeitsbelastung der Einheit	0.33h									
Integer										
Logische Operatoren kennenlernen Integer erklären. Mit Beispielcode an einem Praxisbeispiel Funktion aufzeigen. • Erstellen von integer • Klasse prüfen • von doubles abgrenzen können	5 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein

Gesamtarbeitsbelastung der Einheit	0.08h									
Double										
Double kennenlernen Double kennenlernen • Grundlagen erklären • Einsatzmöglichkeiten • Klasse überprüfen • fehlende Werte (`NaN`) einführen • Sonderfälle erklären	5 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
Double üben Übungsaufgaben, in denen Lernende mit Doubles arbeiten müssen.	15 min	Übung	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein
Gesamtarbeitsbelastung der Einheit	0.33h									
Character										
Character kennenlernen Double kennenlernen • Grundlagen erklären (Nutzung von Anführungszeichen) • Einsatzmöglichkeiten (als Daten (Text) oder auch in R (Farbenfür Grafiken etc.)) • Klasse überprüfen • erste Packages einfließen lassen	15 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
Character üben Übungsaufgaben, in denen Lernende mit Characters arbeiten müssen. • Code komplett neuschreiben • Code anpassen	10 min	Übung	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein

Gesamtarbeitsbelastung der Einheit	0.41h									
Fehlende Werte										
Fehlende Werte kennenlernen Verschiedene fehlende Werte kennenlernen • `NaN` • `NA` • `NULL` • `logical(0)` • `integer(0)` • `double(0)` Allgemeines zu fehlenden Werten: • Länge des Objekts überprüfen • Fehlende Werte in Rechenoperationen • auf fehlende Werte prüfen Beispielcode!	20 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
Umgang mit fehlenden Werten üben Übungsaufgaben, in denen Lernende mit fehlenden Werten arbeiten müssen. • Fehler im Code korrigieren • Multiple Choice zu allgemeinen (logischen) Fragen zu fehlenden Werten	15 min	Übung	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein
Gesamtarbeitsbelastung der Einheit	0.58h									

Datenstrukturen 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%)**										
Variablen										
Variabeln kennenlernen Übersicht und Information über Variablen geben. Mit Beispielcode an einem Praxisbeispiel Funktion aufzeigen, Mit Screenshot Aussehen in Umgebung erklären.	15 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
Variable erstellen Variable mit vorgegebenen Werten erstellen.	15 min	Produktion	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein
Gesamtarbeitsbelastung der Einheit	0.5h									
Atomare Vektoren										

Atomare Vektoren erklären Atomare Vektoren erklären • Funktion `c()` vorstellen • Klammern `[]` einführen • Unterschied Skalare und klassische Verktoren Beispielcodes: • Ansteuern mit Klammern zeigen • Ansteuern mit `TRUE` und `FALSE` zeigen • Vektorisierte Berechnungen	10 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
Mit Variablen arbeiten Übungen zu verschiedenen Unterthemen • Variablen erstellen • einzelne Werte in Variablen ersetzen • mit Vektoren rechnen	20 min	Produktion	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein
Gesamtarbeitsbelastung der Einheit	0.5h									
Faktoren										
Atomare Vektoren erklären Faktoren erklären • als kategoriale Variablen • Beispiele geben (z.B. Geschlecht) Beispielcode	15 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
Factors erstellen Übungsaufgabe, um mit Factors zu arbeiten • Factor erstellen • Factor sortieren	15 min	Übung	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein

Gesamtarbeitsbelastung der Einheit	0.5h									
Matrix										
Matrix erklären Matrix einführen • Beschränkungen aufführen (Class) • Klammern [] und Orientierung erklären	15 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
Matrix erstellen Matrixes selbst erstellen	5 min	Produktion	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein
Matrix organisieren Matixes ansteuern • Spalten • Zeilen	5 min	Übung	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein
Gesamtarbeitsbelastung der Einheit	0.41h									
Funktionen Die Studierenden verstehen die R-Syntax (**50%),** Die Studierenden können eigenständig Code verfassen (**50%)**										
Funktionen erstellen										
Funktionen erklären Funktionen erklären: • Struktur • Syntax Beispielcode zeigen	15 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein

Einfache Funktionen erstellen Kleine Funktionen zu vorgegebenen Aufgaben erstellen (z.B. Funktion, die zwei Zahlen addiert)	5 min	Produktion	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
Gesamtarbeitsbelastung der Einheit	0.33h									
R Pakete Die Studierenden können ihre eigenen Daten an vorliegenden Code anpassen (**20%),** Die Studierenden können vorgegebenen (fremden) Code anpassen (**20%)**										
R Pakete										
Variable erstellen Packages erklären • Installation • Verwendung nach Installation • Übersicht über relevanteste Pakete für Studium/ Forschung Beispielcode!	15 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein
Pakete herunterladen Lernende bestimmtes Paket herunterladen und testen lassen	15 min	Übung	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
Gesamtarbeitsbelastung der Einheit	0.5h									

Datenverarbeitung 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%)**										
Datenverarbeitung										
Beispieldatensatz zeigen	5 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
Daten verarbeiten • Daten anzeigen (bestimmte Werte) • erste deskriptiven Statistiken • fehlende Werte anzeigen lassen • fehlende Werte entfernen • Daten ändern	20 min	Übung	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein
Gesamtarbeitsbelastung der Einheit	0.41h									
Daten importieren/exportieren										
Speichertypen erklären (Export) Verschiedene Speicherformate einführen und Vor- und Nachteile erklären (mit Beispielcode!) • .csv • .rda oder RData • .rds	10 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
Daten importieren Beispielcode zum importieren verschiedener Datentypen	10 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein

Gesamtarbeitsbelastung der Einheit	0.33h									
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	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
Gesamtarbeitsbelastung der Einheit	0.25h									
If-else										
If-else-Statements erklären Erklärung zu verschiedenen Conditions geben Beispielcodes zeigen Auch Verknüpfung verschiedener Loops	15 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
If-else-Statements üben Code mit if-else selbst produzieren	20 min	Produktion	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein
If-else-Statements reflektieren If-else vorgeben und über Output nachdenken lassen	15 min	Untersuchung	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein

Gesamtarbeitsbelastung der Einheit	0.83h									
For-Loop										
For-Loops erklären Beispielcodes (Beispiel mit length() zeigen)	15 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
For-Loop erstellen Code mit For-Loop selbst produzieren	15 min	Produktion	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein
Gesamtarbeitsbelastung der Einheit	0.5h									
While-Loop										
While-Loops erklären Beispielcodes	15 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
While-Loop erstellen Code mit While-Loop selbst produzieren	15 min	Produktion	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein
Negativbeispiel bearbeiten Negativbeispiel aufzeigen	10 min	Untersuchung	Online	Asynchron	Lehrer anwesend	Nein	Nein	Nein	Nein	Nein
Gesamtarbeitsbelastung der Einheit	0.66h									
Debugging 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%)**										
Warnungen und Fehler										

Warnungen und Fehler unterscheiden Warnings/Errors einführen • Unterschied Warning/Error • Funktionsweise Error (if-Statement) • eigene Funktionen und Warnungen Beispielcode zeigen • Warnung selbst schreiben • Error selbst schreiben	15 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein
Warnung selbst schreiben Übungsaufgabe, wo Studierende selbst eine simple Warnung schreiben sollen	15 min	Produktion	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein
Tipps Weiterführende Informationen und Tipps rund um Errors/Warnings • Help-Seite von Funktionen • häufige Fehler vermeiden • etc.	15 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein
Gesamtarbeitsbelastung der Einheit	0.75h									
Umgebung(en) Die Studierenden verstehen die R-Syntax (**100%)**										
Funktionen										
Funktionen	20 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
Loops	10 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein
Gesamtarbeitsbelastung der Einheit	0.5h									

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	Untersuchung	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	0	Formativ	Automatisiert
Gesamtarbeitsbelastung der Einheit	0.5h											
Code umschreiben												
Übung Code umschreiben Vorgegebener (funktionierender) Code, der umgeschrieben werden soll, ohne Funktionalität zu verlieren	20 min	Produktion	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein		
Gesamtarbeitsbelastung der Einheit	0.33h											
Datensätze nutzen												
Erstellung eines Datensatzes Code zeigen, der Erstellung eines Datensatzes zeigt	10 min	Erwerb	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Nein	Nein	Nein		

Datensatz-Navigation üben Übungsaufgaben, in denen Studierende lernen sollen, mit einem vorgegebenen Datensatz umzugehen • Fehlende Werte • bestimmte Zeilen ausgeben lassen	20 min	Produktion	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein
Gesamtarbeitsbelastung der Einheit	0.5h									
Komplexe Programmierübungen										
allgemeine Programmierübungen	120 min	Produktion	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein
Gesamtarbeitsbelastung der Einheit	2h									
Freies Üben Die Studierenden können eigenständig Code verfassen (**100%)**										
Code Chunk										
Code Chunk Leerer Codechunk (ähnlich in RStudio), wo frei gecodet werden kann	240 min	Produktion	Online	Asynchron	Lehrer nicht anwesend	Nein	Nein	Automatisiert	Nein	Nein
Gesamtarbeitsbelastung der Einheit	4h									
Gesamtarbeitsbelastung des Kurses	**16.33h**									