

SER 321 B Session

SI Session

Sunday, October 27th 2024

7:00 pm - 8:00 pm MST

Agenda



HTTP

Requests

Responses

JSON

SI Session Expectations

Thanks for coming to the **SER 321** SI session. We have a packed agenda and we are going to try to get through as many of our planned example problems as possible. This session will be recorded and shared with others.

- If after this you want to see additional examples, please visit the drop-in tutoring center.
- We will post the link in the chat now and at the end of the session.
 - tutoring.asu.edu
- Please keep in mind we are recording this session and it will be made available for you to review 24-48 hours after this session concludes.
- Finally, please be respectful to each other during the session.

Interact with us:

Zoom Features



Zoom Chat

- Use the chat feature to interact with the presenter and respond to presenter's questions.
- Annotations are encouraged

SER 321

HTTP(S)



Stateful

OR

Stateless

Synchronous

OR

Asynchronous

Check out the recording for the discussion!

Check out the recording for the discussion!

We will focus on ***FOUR*** request types

1.

2.

3.

4.

SER 321

HTTP Requests

Check out the recording for the discussion!

The difference between `PUT` and `POST` is that `PUT` is idempotent: calling it once is no different from calling it several times successively (there are no *side effects*).

1. GET

2. POST

3. PUT

4. DELETE

Idempotent

SER 321

HTTP Responses

Check out the recording for the discussion!

Status Codes →

1XX

2XX

3XX

4XX

5XX

SER 321

HTTP Responses

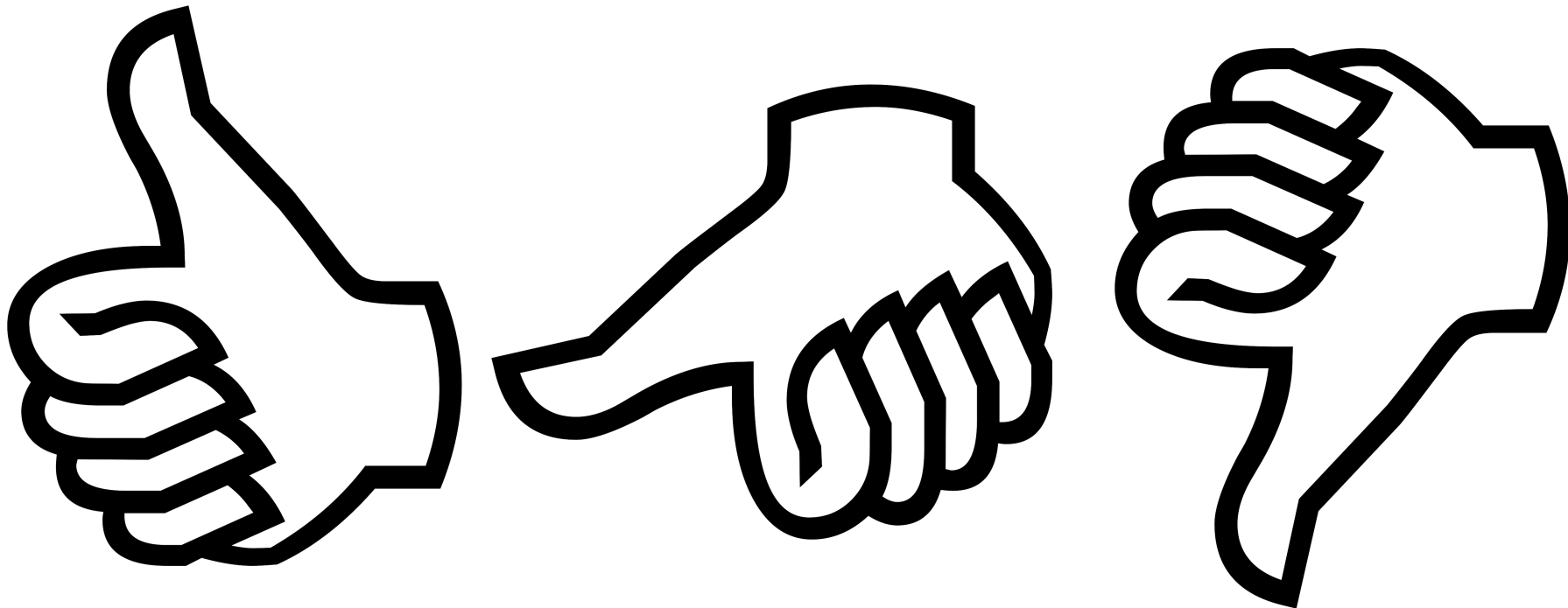
Check out the recording for the discussion!

1XX	Information	Rare			
2XX	Success	200	201	204	
3XX	Redirect	301			
4XX	User Error	400	403	404	405
5XX	Server Error	501	503		

SER 321

JSON

How comfortable are we with JSON?



SER 321

JSON

Object

Array

Member

{ }

[]

“name” : value

SER 321

JSON

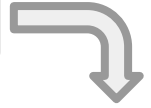
Data is stored in...

Name:Value pairs

AKA

Members

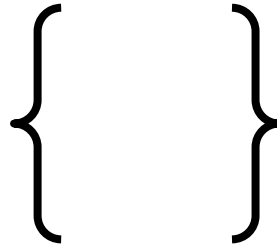
"Katie"



"student" : "Katie"

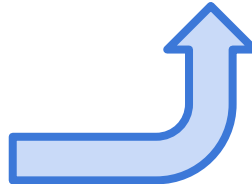
What uses curly braces?

Objects



What do Objects contain?

Members



SER 321

JSON

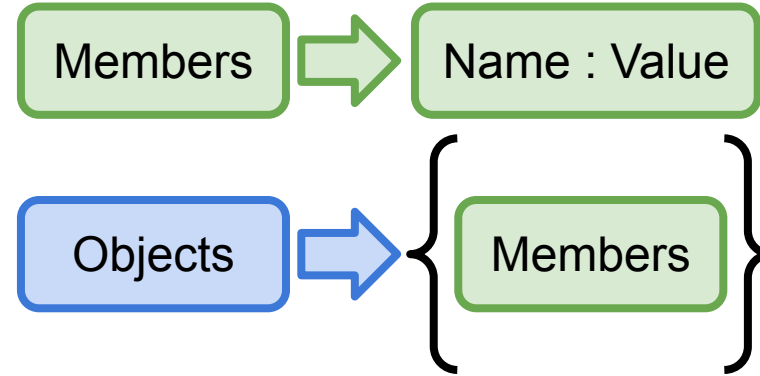
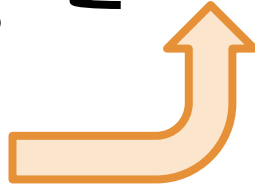
What uses brackets?

Arrays

[]

What do Arrays contain?

Any ***Valid*** Value



SER 321

JSON

What is a valid value?

Strings

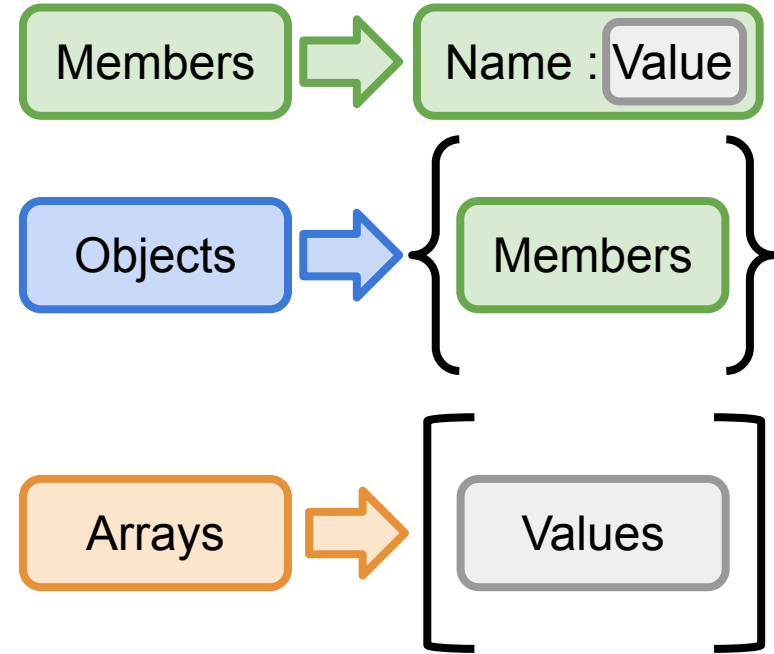
Booleans

Numbers

NULL

Objects

Arrays



Check out the recording for the discussion!

SER 321

JSON

What do we see?

```
{  
  "name" : "katie",  
  "role" : "student",  
  "course" : "ser321"  
}
```

```
[  
  {"name" : "katie"}  
]
```

Check out the recording for the discussion!

```
{  
  "name" : "katie",  
  "pets" : {  
    "dog" : "smokey"  
  }  
}
```

```
{  
  "submissions" : [  
    {"name" : "katie"},  
    {"name" : "zac"}  
  ]  
}
```

SER 321

JSON

Do you see any errors?

```
1. {  
    "name" : "katie",  
    "role" : "student",  
    "course" : "ser321"  
}
```

```
2. [{  
    "name" : "katie",  
    "name" : "zac"  
}]
```

Check out the recording for the discussion!

```
3. {  
    "name" : "katie",  
    "pets" : {  
        "dog" : "smokey",  
        "dog" : "samie"  
    }  
}
```

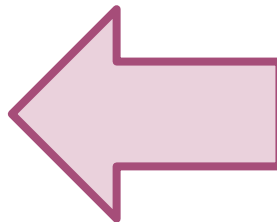
```
4. {  
    "submissions" : [  
        {"name" : "katie"},  
        {"name" : "zac"}  
    ]  
}
```


SER 321

JSON

Can we fix these?

```
[  
  {"name": "katie"},  
  {"name": "zac"}  
]
```



```
[{  
  "name": "katie",  
  "name": "zac"  
}]
```

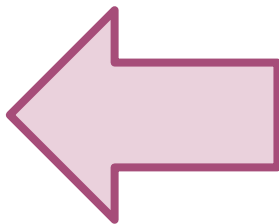
WRONG

SER 321

JSON

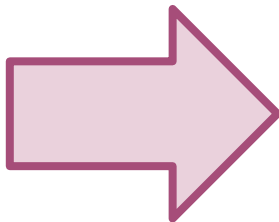
Can we fix these?

```
[  
  {"name": "katie"},  
  {"name": "zac"}  
]
```



```
[{  
  "name": "katie",  
  "name": "zac"  
}]  
WRONG
```

```
{  
  "name": "katie",  
  "pets": {  
    "dog": "smokey",  
    "dog": "samie"  
  }  
}  
WRONG
```



```
{  
  "name": "katie",  
  "pets": {  
    {"dog": "smokey"},  
    {"dog": "samie"}  
  }  
}
```

SER 321

JSON in Assignments

Starter Code files use **org.json**

If you have a different JSON library you would prefer to use, you *can* use that

This library is **very** intuitive, so I recommend using it!

JSONObject

A JSONObject is an unordered collection of name/value pairs.

JSONArray

A JSONArray is an ordered sequence of values.

SER 321

JSON in Assignments

Very Similar to a **MAP** object

JSONObject

A JSONObject is an unordered collection of name/value pairs.

boolean

`has(String key)`

Determine if the JSONObject contains a specific key.

```
boolean hasStu = json.has("Student");
```

SER 321

JSONObject	<code>put(String key, boolean value)</code> Put a key/boolean pair in the JSONObject.
JSONObject	<code>put(String key, Collection<?> value)</code> Put a key/value pair in the JSONObject, where the value will be a JSONArray which is produced from a Collection.
JSONObject	<code>put(String key, double value)</code> Put a key/double pair in the JSONObject.
JSONObject	<code>put(String key, float value)</code> Put a key/float pair in the JSONObject.
JSONObject	<code>put(String key, int value)</code> Put a key/int pair in the JSONObject.
JSONObject	<code>put(String key, long value)</code> Put a key/long pair in the JSONObject.
JSONObject	<code>put(String key, Map<?,?> value)</code> Put a key/value pair in the JSONObject, where the value will be a JSONObject which is produced from a Map.
JSONObject	<code>put(String key, Object value)</code> Put a key/value pair in the JSONObject.

Put will add to the Object

```
json.put("term", "fall");
```

SER 321

JSON in Assignments

Need to know the data type
when pulling from the
JSONObject!

```
JSONObject json = {  
    "Student" : "katie",  
    "term" : "fall"  
}
```

```
String term = json.get("term");
```



```
String term = json.getString("term");
```



Object	<code>get(String key)</code> Get the value object associated with a key.
BigDecimal	<code>getBigDecimal(String key)</code> Get the BigDecimal value associated with a key.
BigInteger	<code>getBigInteger(String key)</code> Get the BigInteger value associated with a key.
boolean	<code>getBoolean(String key)</code> Get the boolean value associated with a key.
double	<code>getDouble(String key)</code> Get the double value associated with a key.
<code><E extends Enum<E>></code> E	<code>getEnum(Class<E> clazz, String key)</code> Get the enum value associated with a key.
float	<code>getFloat(String key)</code> Get the float value associated with a key.
int	<code>getInt(String key)</code> Get the int value associated with a key.
JSONArray	<code>getJSONArray(String key)</code> Get the JSONArray value associated with a key.
JSONObject	<code>getJSONObject(String key)</code> Get the JSONObject value associated with a key.
long	<code>getLong(String key)</code> Get the long value associated with a key.
static String[]	<code>getNames(JSONObject jo)</code> Get an array of field names from a JSONObject.
static String[]	<code>getNames(Object object)</code> Get an array of public field names from an Object.
	<code>getNumber(String key)</code> Get the Number value associated with a key.
	<code>getString(String key)</code> Get the string associated with a key.

SER 321

JSON in Assignments

JSONArray

No HAS

Can target indices

Object	<code>get(int index)</code> Get the object value associated with an index.
BigDecimal	<code>getBigDecimal(int index)</code> Get the BigDecimal value associated with an index.
BigInteger	<code>getBigInteger(int index)</code> Get the BigInteger value associated with an index.
boolean	<code>getBoolean(int index)</code> Get the boolean value associated with an index.
double	<code>getDouble(int index)</code> Get the double value associated with an index.
<code><E extends Enum<E>> E</code>	<code>getEnum(Class<E> clazz, int index)</code> Get the enum value associated with an index.
float	<code>getFloat(int index)</code> Get the float value associated with a key.
int	<code>getInt(int index)</code> Get the int value associated with an index.
JSONArray	<code>getJSONArray(int index)</code> Get the JSONArray associated with an index.
JSONObject	<code>getJSONObject(int index)</code> Get the JSONObject associated with an index.
long	<code>getLong(int index)</code> Get the long value associated with an index.
Number	<code>getNumber(int index)</code> Get the Number value associated with a key.
String	<code>getString(int index)</code> Get the string associated with an index.

SER 321

Scratch Space

Upcoming Events

SI Sessions:

- Tuesday, October 29th 2024 at 10:00 am MST
- Thursday, October 31st 2024 at 7:00 pm MST
- Sunday, November 3rd 2024 at 7:00 pm MST



Review Sessions:

- TBD

Don't forget about
Daylight Savings!

Questions?

Survey:

<https://asuasn.info/ASNSurvey>



More Questions?

Check out our other resources!

tutoring.asu.edu



Academic Support

Academic Support Network (ASN) provides a variety of free services in-person and online to help currently enrolled ASU students succeed academically.

Services



Subject Area Tutoring

Need in-person or online help with math, science, business, or engineering courses? Just hop into our Zoom room or drop into a center for small group tutoring. We'll take it from there.

[Need help using Zoom?](#)

[View the tutoring schedule](#)

[View digital resources](#)

Go to Zoom



Writing Tutoring

Need help with undergraduate or graduate writing assignments? Schedule an in-person or online appointment, access your appointment link, or wait in our drop-in queue.

[Access your appointment link](#)

[Access the drop-in queue](#)

Schedule Appointment



Online Study Hub

Join our online peer communities to connect with your fellow Sun Devils. Engage with our tools to search our bank of resources, videos, and previously asked questions. Or, ask our Tutorbot questions.

Now supporting courses in Math, Science, Business, Engineering, and Writing.

Online Study Hub

1-

Go to Zoom

2-

[Need help using Zoom?](#)

[View the tutoring schedule](#)

[View digital resources](#)



1. Click on 'Go to Zoom' to log onto our Online Tutoring Center.
2. Click on 'View the tutoring schedule' to see when tutors are available for specific courses.

More Questions?

Check out our other resources!

tutoring.asu.edu/online-study-hub

 **Academic Support Network**

 [Services](#)  [Faculty and Staff Resources](#) [About Us](#) 

[University College](#)

Online Study Hub

Online peer communities for students and tutors, YouTube channels, and Tutorbots.



What are online peer communities?

Individual courses have an online peer community that allows you to connect with your peers to post and answer questions and to develop study groups.



How can tutoring center videos help?

Videos can help supplement the learning you're doing in and outside of class and include step-by-step methods for how to understand concepts.



How does the Tutorbot work?

You can ask the Tutorbot questions about course concepts and the Tutorbot will recommend additional resources and examples to help address your questions.

Select a subject

- Any -

[Apply](#)



Academic Support Network



[Services](#) 

[Faculty and Staff Resources](#)

[About Us](#) 

[University College](#)

Select a subject

- Any -

[Apply](#)

Business

ACC 231

Uses of Accounting Info I

 [Peer Community](#)

ACC 241

Uses of Accounting Info II

 [Peer Community](#)

CIS 105

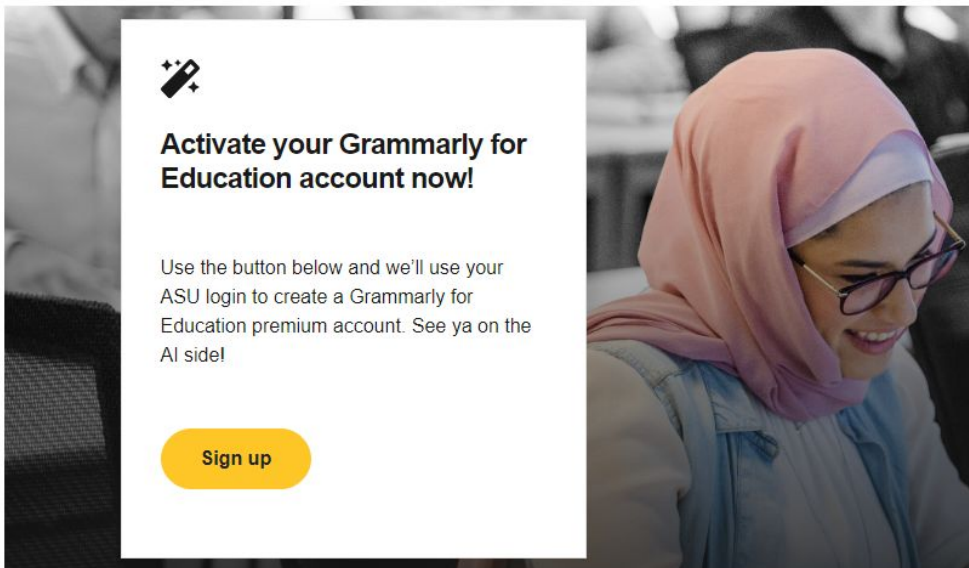
Computer Applications and Information Technology


 [Peer Community](#)

Don't forget to check out the Online Study Hub for additional resources!

Expanded Writing Support Available

Including Grammarly for Education, at no cost!





Activate your Grammarly for Education account now!

Use the button below and we'll use your ASU login to create a Grammarly for Education premium account. See ya on the AI side!

[Sign up](#)



tutoring.asu.edu/expanded-writing-support

*Available slots for this pilot are limited

Additional Resources

- [Course Repo](#)
- [Gradle Documentation](#)
- [GitHub SSH Help](#)
- [Linux Man Pages](#)
- [OSI Interactive](#)
- [MDN HTTP Docs](#)
 - [Requests](#)
 - [Responses](#)
- [JSON Guide](#)
- [org.json Docs](#)