

Task:

Choose 3 EPAM sites and analyze it using DevTools console:

- Do they have memory leaks?
- What functions are taking more time for execution?
- What time does it take for rendering first meaningful page?

1. EPAM Learn - <https://learn.epam.com/start>

1.1 This site does not have significant memory leaks, since the memory size after some activities (Snapshot 2) on the same page is less than memory before (Snapshot 1).

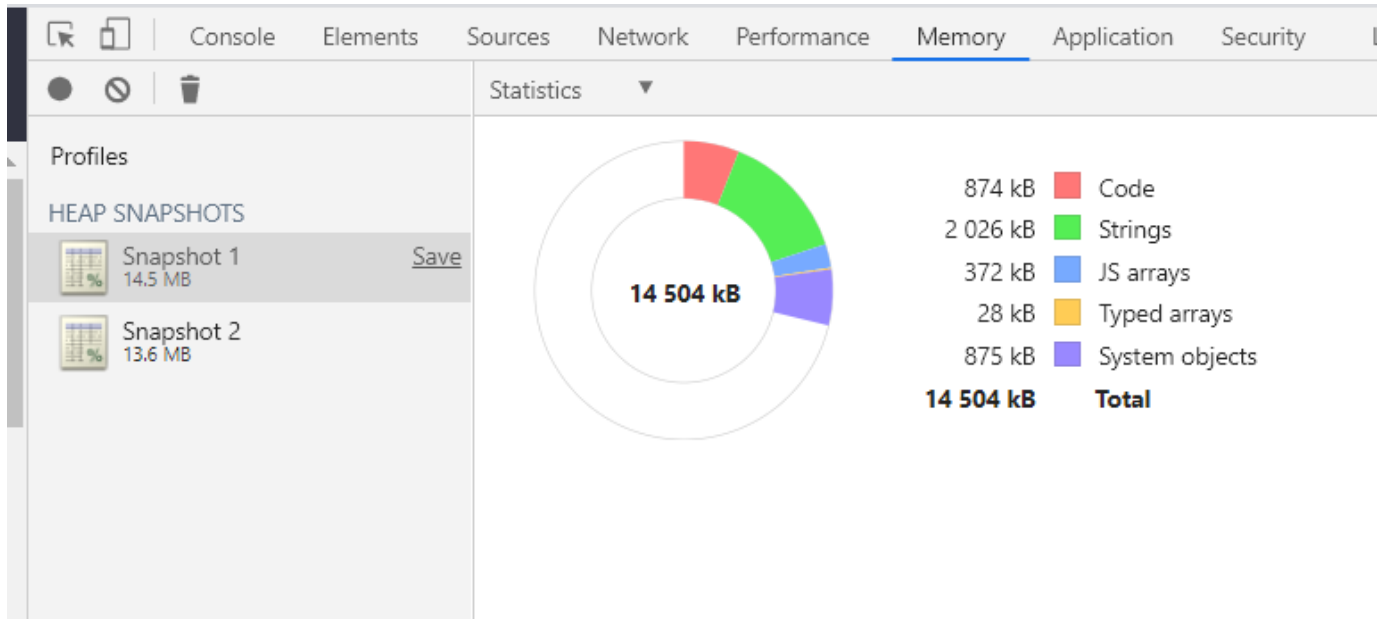
The screenshot shows the EPAM Learn website in a browser window. The website has a dark blue header with the 'LEARN Digital Platform' logo. The main content area features a 'Just keep going!' message, a progress bar for 'SwitchJS 2021' (24% completed), and a search bar. Below this, there are 'Top recommended for you' and a list of courses including 'Face-To-Face, Online' and 'ENG'.

Overlaid on the right side of the browser window is the Chrome DevTools Memory tab. The 'HEAP SNAPSHOTS' panel shows two snapshots: 'Snapshot 1' (14.5 MB) and 'Snapshot 2' (13.6 MB). The 'Comparison' panel is open, showing a table of memory usage for Snapshot 1.

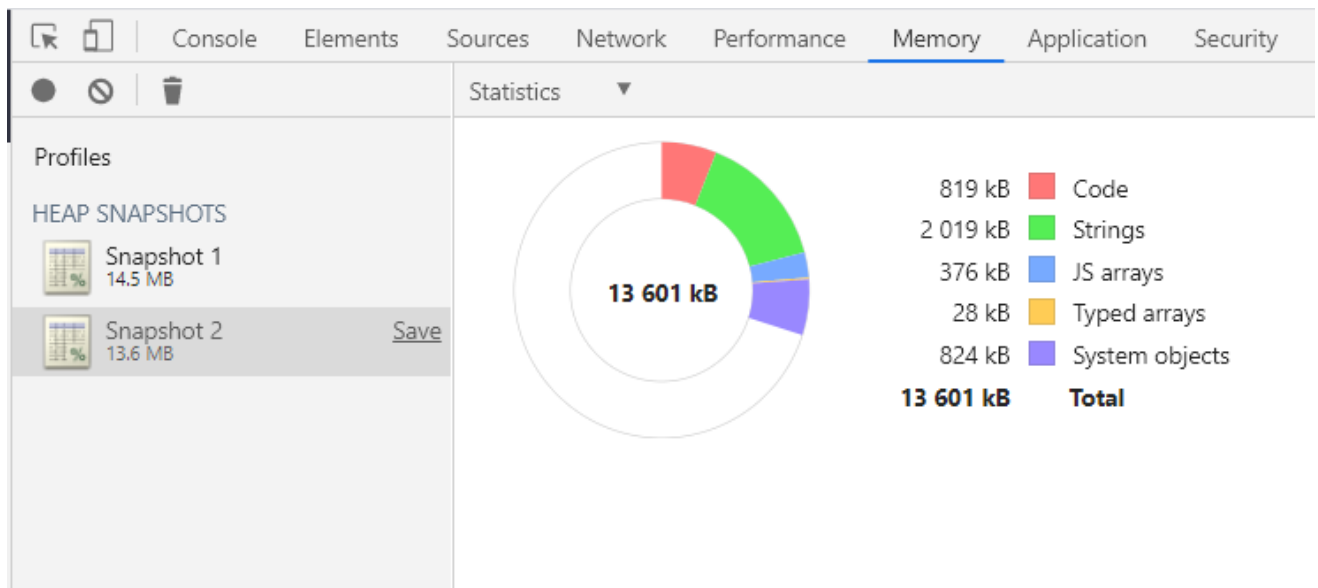
Constructor	# New	# Deleted	# Delta	Alloc. Size	Freed Size	Size Delta
(system)	130 806	147 190	-16 384	4 925 488	5 360 196	-434 708
(array)	45 647	46 303	-656	3 016 844	3 278 772	-261 928
(string)	35 364	35 838	-474	1 800 504	1 810 736	-10 232
(closure)	24 453	29 097	-4 644	749 092	883 744	-134 652
Object	25 542	25 154	+388	733 876	720 768	+13 108
(compiled code)	15 698	16 867	-1 169	683 720	739 868	-56 148
Array	13 639	13 589	+50	218 560	217 760	+800
Fs	1 981	1 815	+166	206 024	188 760	+17 264
system / Context	2 988	3 351	-363	92 220	101 688	-9 468
(concatenated string)	4 209	4 623	-414	84 180	92 460	-8 280
e	1 220	1 302	-82	29 284	34 416	-5 132
u	505	505	0	28 280	28 280	0
Map	960	969	-9	15 408	15 600	-192
(regexp)	466	476	-10	13 048	13 328	-280
t	320	381	-61	12 092	14 144	-2 052
(sliced string)	523	526	-3	10 460	10 520	-60
Set	479	477	+2	7 760	7 776	-16
b	105	105	0	4 836	4 836	0
(number)	358	343	+15	4 296	4 116	+180
	70	72	-2	3 606	3 884	-278

The 'Retainers' panel shows a single entry for 'Object' with columns for 'Distance', 'Shallow Size', and 'Retained Size'.

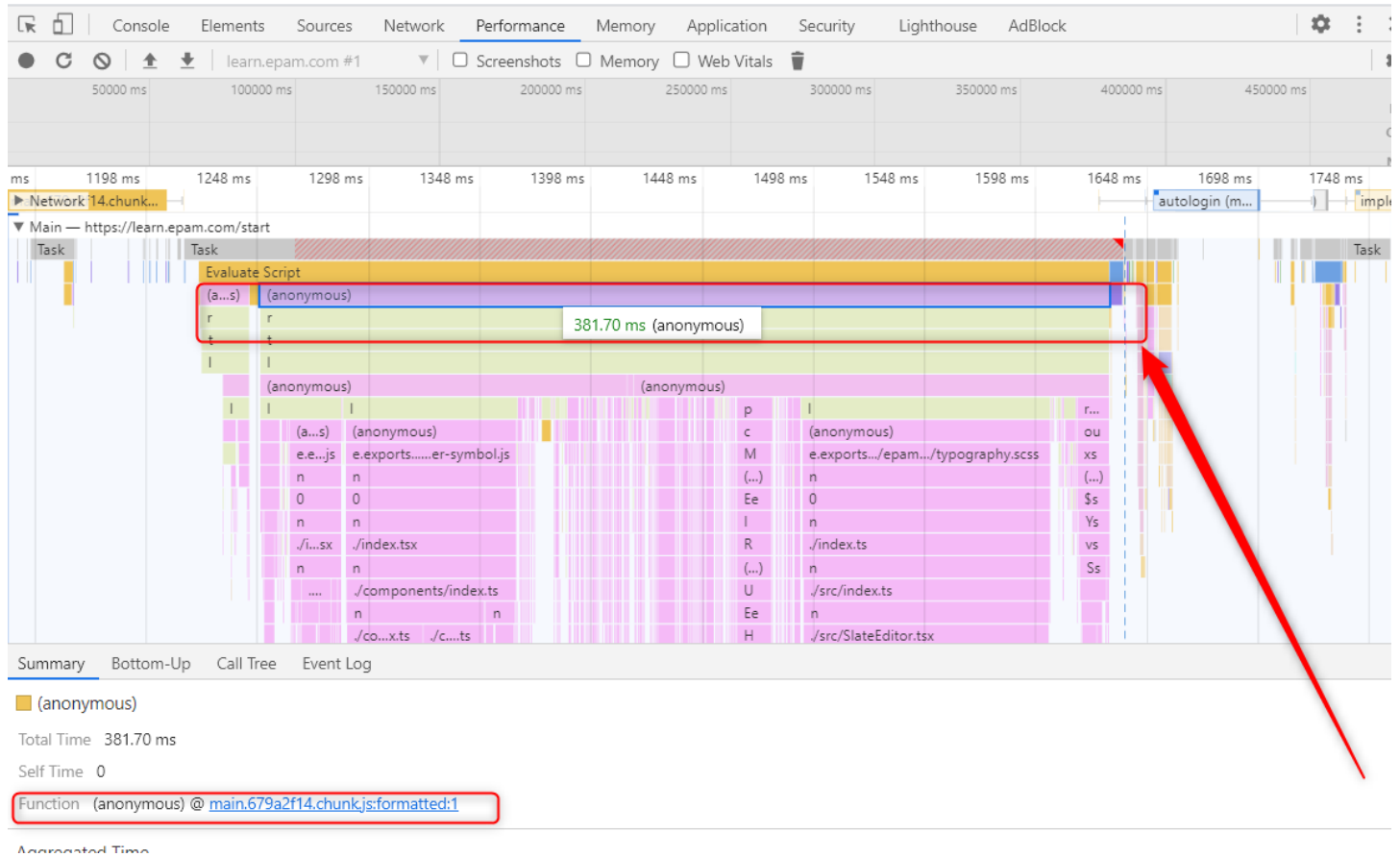
Snapshot 1



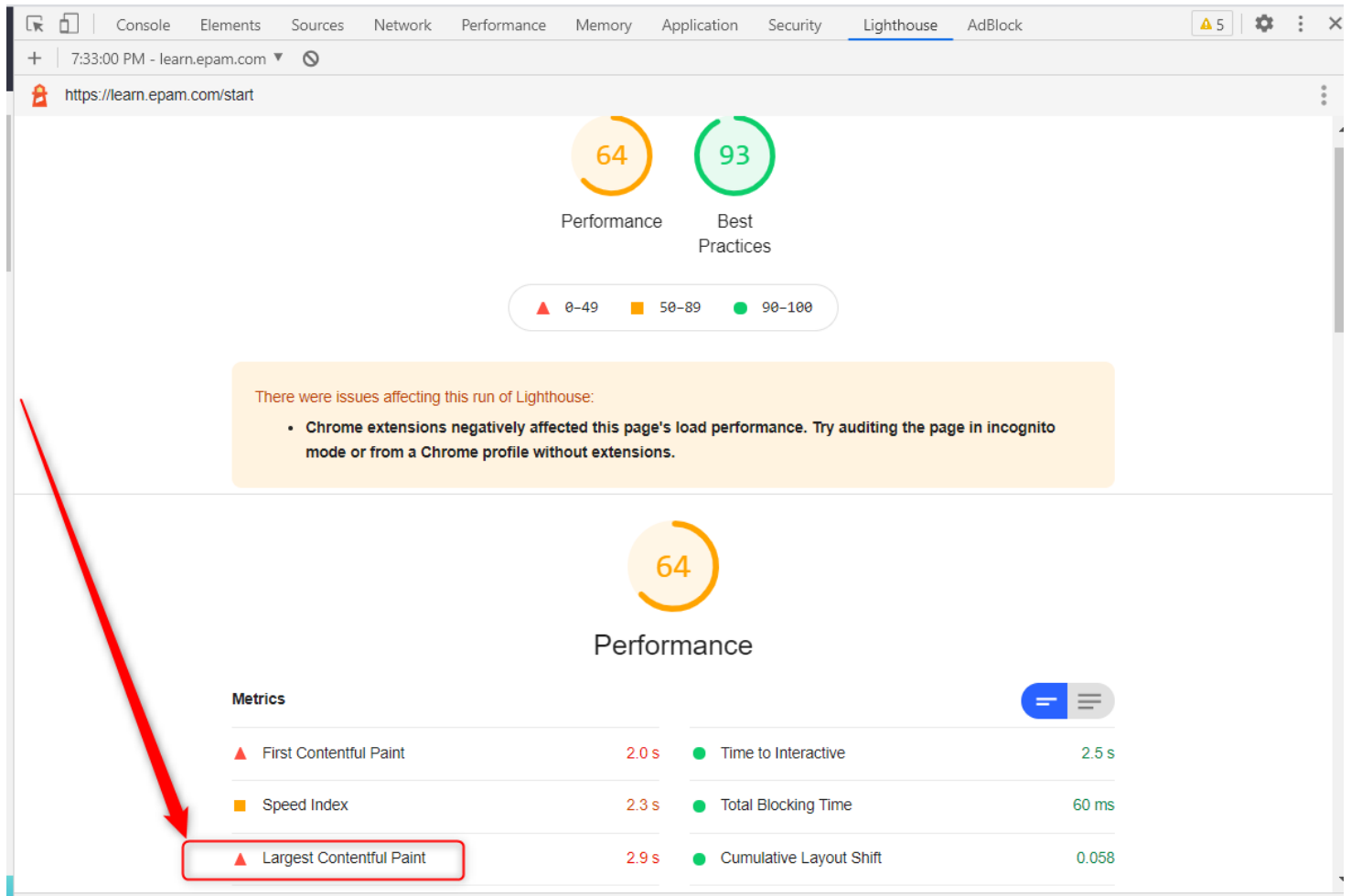
Snapshot 2



1.2 The longest function takes **381.70ms** and this is anonymous function.



1.3 As First Meaningful Paint (FMP) is deprecated in Lighthouse 6.0, the most suitable metric is Largest Contentful Paint (LCP) instead. **LCP = 2.9 s.**



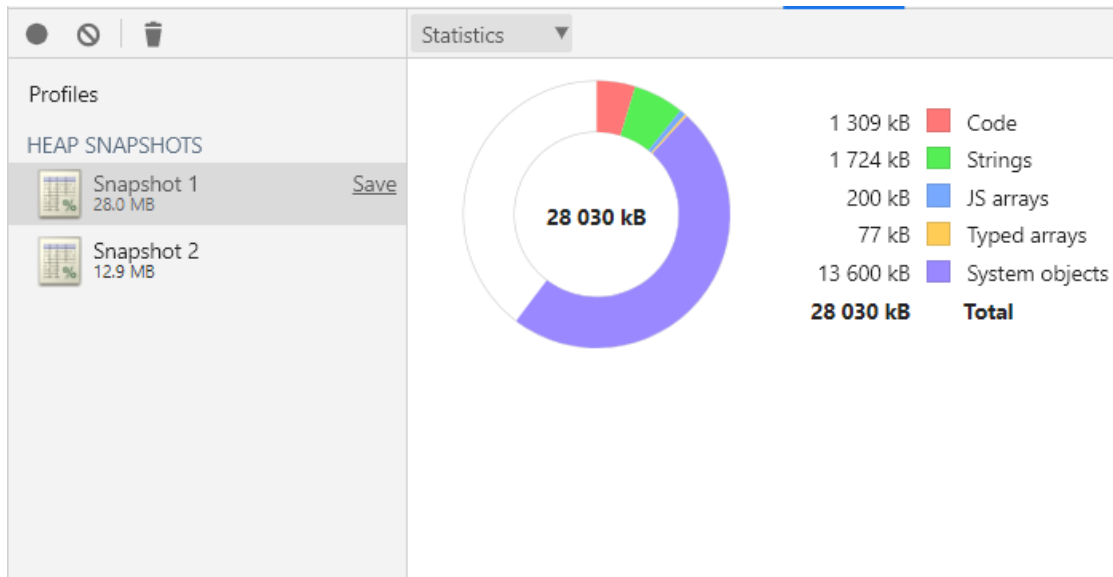
2. EPAM Grow - <https://grow.epam.com>

2.1 This site does not have significant memory leaks since the memory size after some activities (Snapshot 2) on the same page is less than memory before (Snapshot 1).

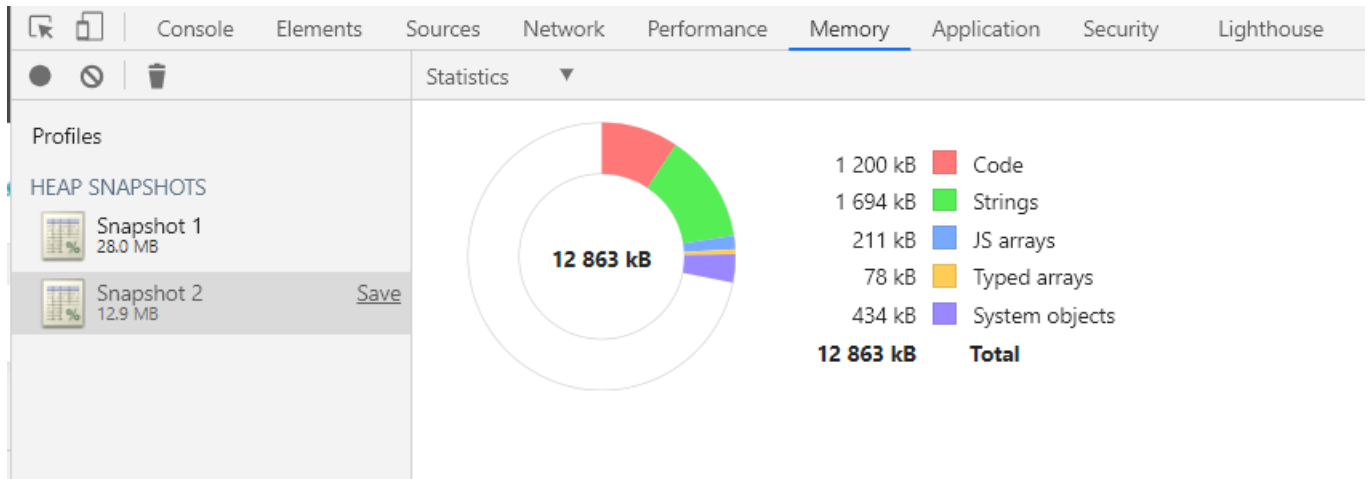
The screenshot shows the EPAM Grow website in a web browser. The left sidebar contains navigation links: ME, ASSESSMENT, and MORE. The main content area displays a list of activities, including 'Testing L2, Functional testing' and 'Testing L2, Functional Testing'. The right sidebar shows the Chrome DevTools Memory tab, which is open to the 'Comparison' view. The 'Class filter' is set to 'Snapshot 1'. The table below shows the memory usage for various objects, comparing Snapshot 1 and Snapshot 2. The 'Size Delta' column indicates the change in memory size between the two snapshots.

Constructor	# New	# Deleted	# Delta	Alloc. Size	Freed Size	Size Delta
(system)	127 943	235 418	-107 475	3 721 092	7 438 944	-3 717 852
(array)	21 316	67 457	-46 141	2 393 296	7 779 152	-5 385 856
(closure)	48 259	94 226	-45 967	1 512 960	2 961 200	-1 448 240
(string)	38 081	55 450	-17 369	1 224 508	1 911 664	-687 156
(compiled code)	19 294	41 185	-21 891	1 116 500	2 409 776	-1 293 276
Object	22 892	50 814	-27 922	681 368	1 479 116	-797 748
system / Context	22 529	45 134	-22 605	505 992	1 013 072	-507 080
ae	4 821	10 039	-5 218	443 532	923 588	-480 056
(concatenated string)	21 407	41 724	-20 317	428 140	834 480	-406 340
Array	5 210	11 678	-6 468	83 528	187 040	-103 512
e	802	1 696	-894	70 076	154 312	-84 236
HTMLDivElement	1 883	1 897	-14	52 528	52 864	-336
t	1 007	2 164	-1 157	47 684	102 784	-55 100
(regexp)	584	1 210	-626	16 352	33 880	-17 528
i	346	743	-397	15 964	34 332	-18 368
n	361	802	-441	14 084	31 704	-17 620
Date	248	564	-316	11 904	27 072	-15 168
(sliced string)	356	766	-410	7 120	15 320	-8 200
(number)	571	1 317	-746	6 852	15 804	-8 952
Retainers						
Object				Distance	Shallow Size	Retained Size

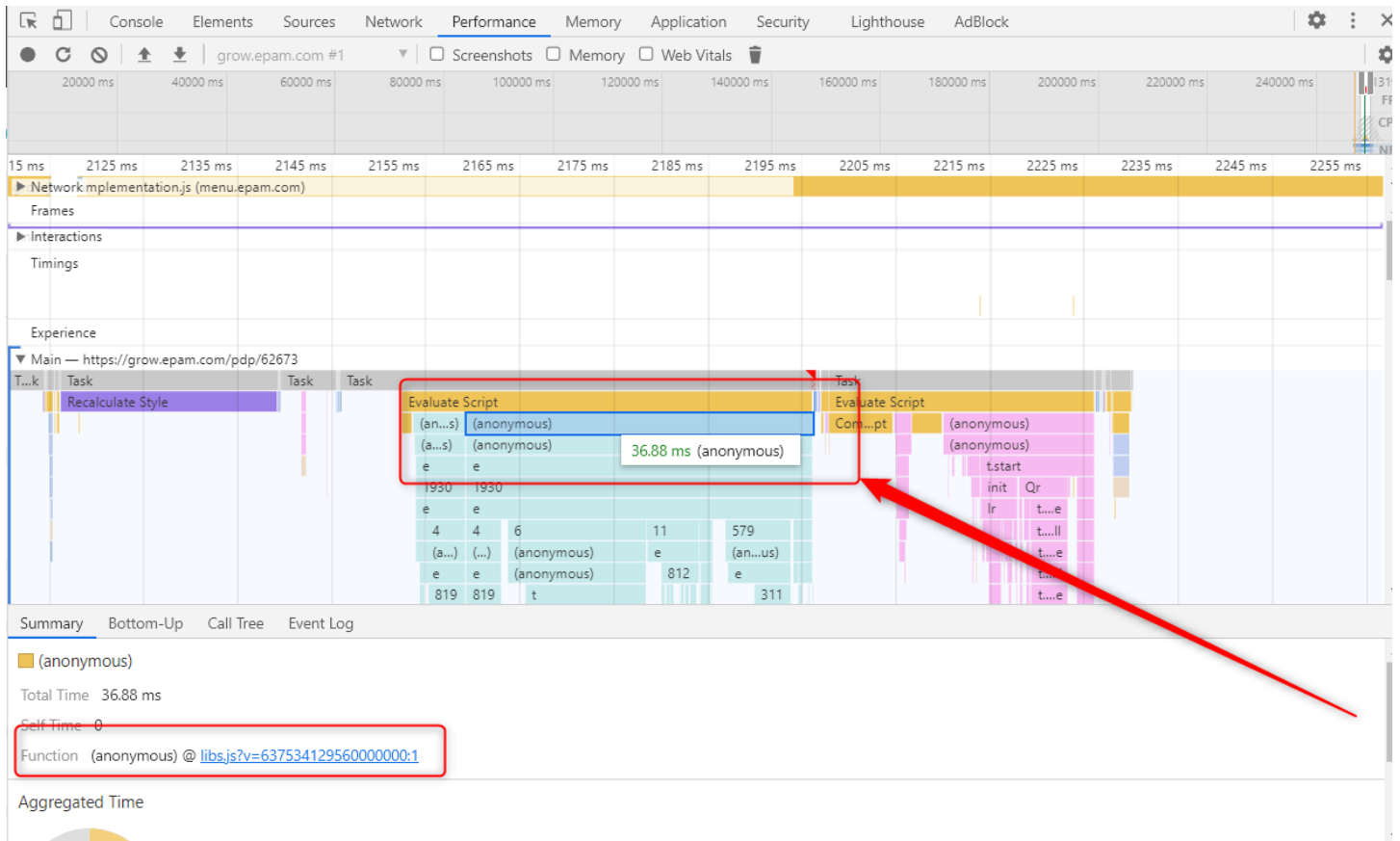
Snapshot 1



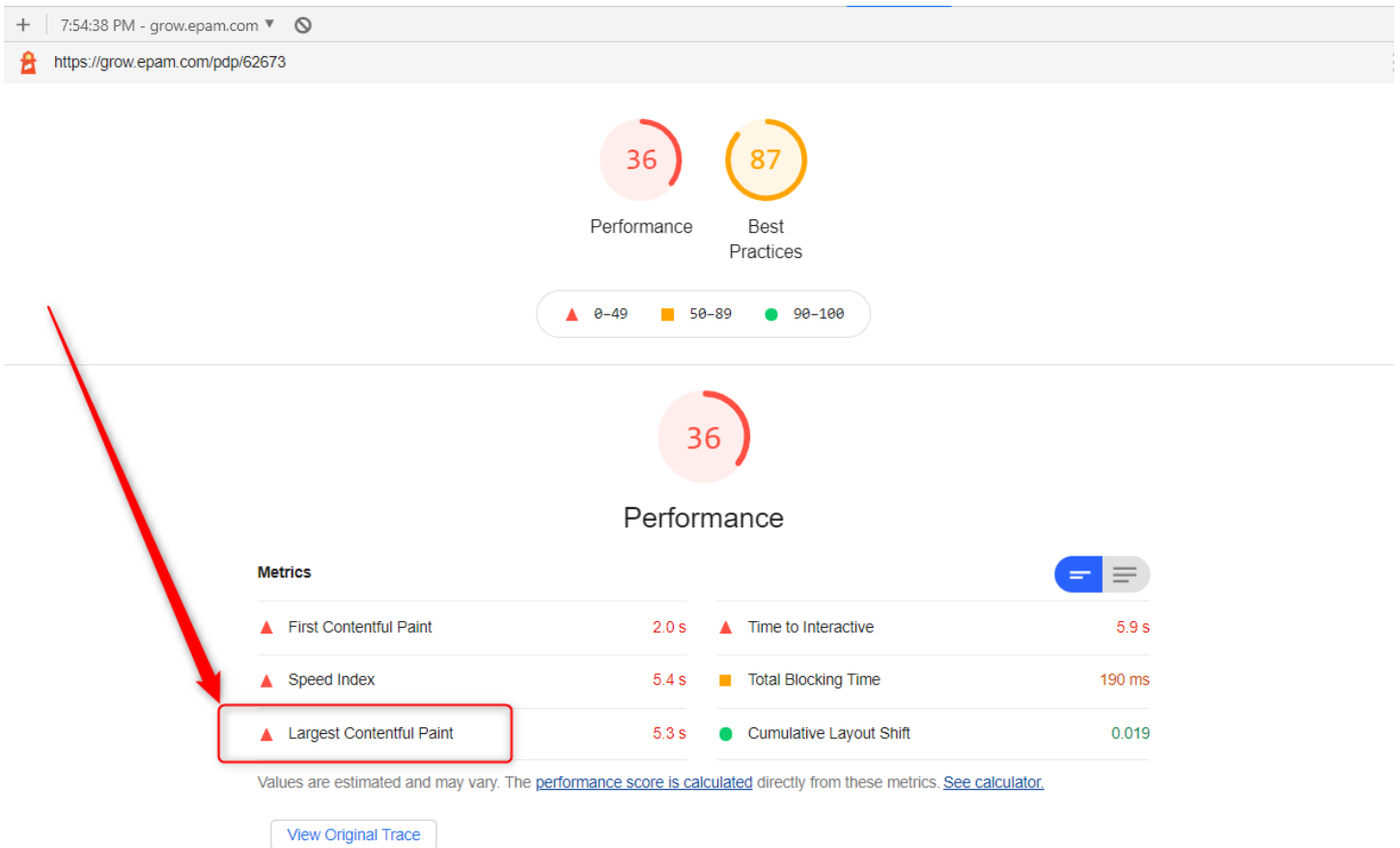
Snapshot 2



2.2 The longest function takes 36.88ms and this is anonymous function.



2.3 LCP = 5.3 s.



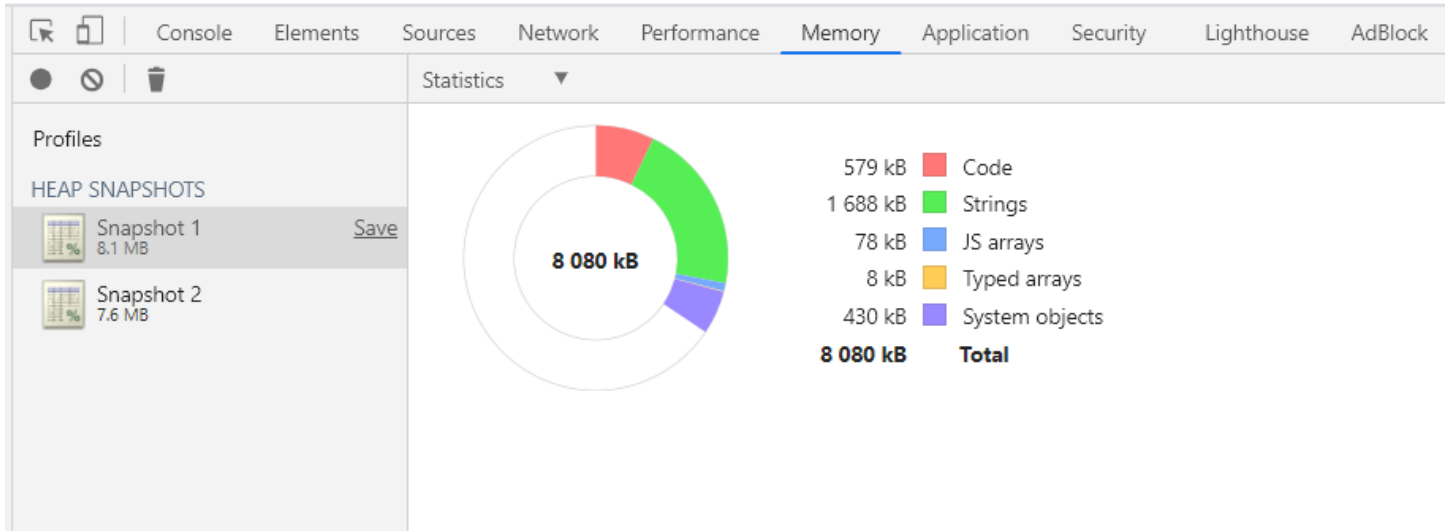
3. EPAM Vacation - <https://vacation.epam.com/>

3.3 This site does not have significant memory leaks since the memory size after some activities (Snapshot 2) on the same page is less than memory before (Snapshot 1).

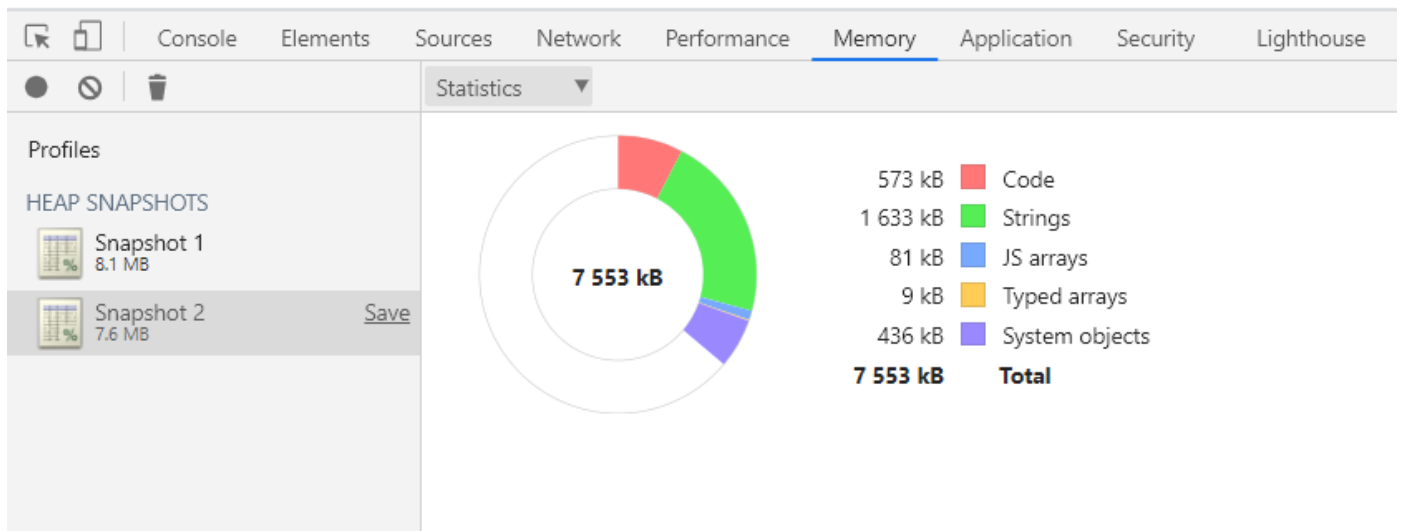
The screenshot shows the EPAM Vacation website on the left and the Chrome DevTools Memory tab on the right. The website displays a 'REMAINING BALANCE' of 15d 0h and a 'Vacation Days' counter with a red '1' icon. The DevTools Memory tab shows a comparison between Snapshot 1 (8.1 MB) and Snapshot 2 (7.6 MB). The table below lists the memory usage for various constructors and retainers.

Constructor	# New	# Deleted	# Delta	Alloc. Size	Freed Size	Size Delta
(system)	89 970	92 730	-2 760	2 505 020	2 805 616	-300 596
(array)	7 144	12 431	-5 287	1 373 580	1 554 208	-180 628
(string)	15 053	15 674	-621	697 652	742 356	-44 704
(closure)	21 014	20 949	+65	633 732	631 664	+2 068
(compiled code)	9 871	10 212	-341	526 392	528 584	-2 192
(concatenated string)	23 081	23 375	-294	461 620	467 500	-5 880
Object	2 758	2 722	+36	79 892	79 048	+844
Array	1 270	1 251	+19	21 016	20 472	+544
system / Context	527	482	+45	12 900	11 684	+1 216
HTMLDivElement	99	98	+1	2 044	1 848	+196
(regexp)	36	32	+4	1 008	896	+112
HTMLCollection	133	132	+1	840	784	+56
HTMLLIElement	84	84	0	812	588	+224
e	15	15	0	812	812	0
HTMLAnchorElement	105	101	+4	756	504	+252
DOMTokenList	189	176	+13	728	252	+476
NamedNodeMap	29	22	+7	644	420	+224
(number)	38	70	-32	456	840	-384
HTMLElement	25	25	0	448	392	+56
HTMLButtonElement	16	16	0	470	364	+106
Retainers						
Object				Distance	Shallow Size	Retained Size

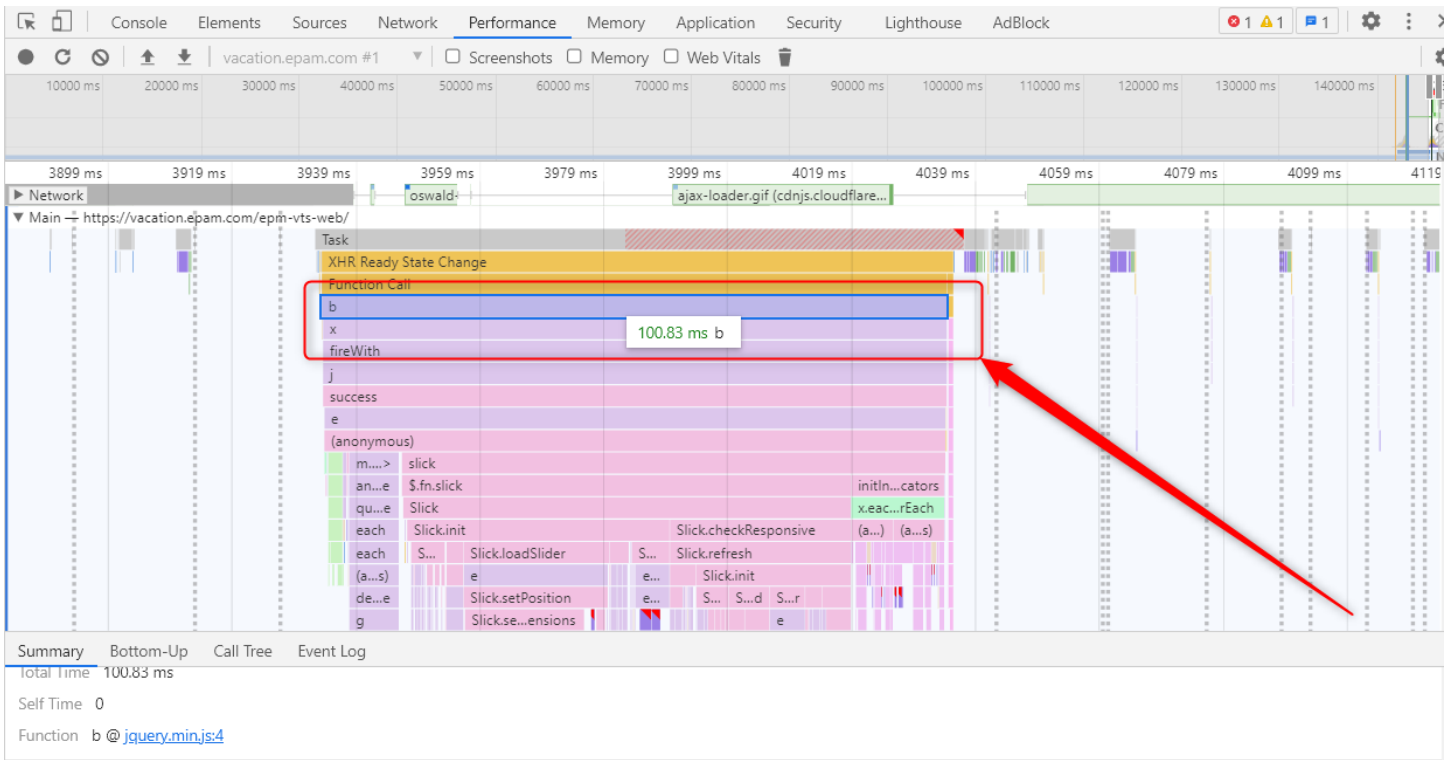
Snapshot 1



Snapshot 2



3.2 The longest function takes 100.83ms and this is anonymous function.



3.3 LCP = 5.1 s.

